AD-E800931

(2)

AD-A141 811

AFATL-TR-83-42

Transonic Interference Flow-Field Analysis for Wing-Body-Pylon-Store Configurations

P Sundaram
S N Chaudhuri

THE UNIVERSITY OF TENNESSEE SPACE INSTITUTE
TULLAHOMA, TENNESSEE 37388

MAY 1983

FINAL REPORT FOR PERIOD SEPTEMBER 1981–SEPTEMBER 1982

Approved for public release; distribution unlimited

Air Force Armament Laboratory
AIR FORCE SYSTEMS COMMAND★UNITED STATES AIR FORCE★EGLIN AIR FORCE BASE, FLORIDA

NOTICE

Please do not request copies of this report from the Air Force Armament Laboratory. Additional copies may be purchased from:

National Technical Information Service
5285 Port Royal Road
Springfield, Virginia 22161

Federal Government agencies and their contractors registered with Defense Technical Information Center should direct requests for copies of this report to:

Defense Technical Information Center
Cameron Station
Alexandria, Virginia 22314

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFATL-TR-83-42	2. GOVT ACCESSION NO. A [illegible]	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) TRANSONIC INTERFERENCE FLOW-FIELD ANALYSIS FOR WING-BODY-PYLON-STORE CONFIGURATIONS		5. TYPE OF REPORT & PERIOD COVERED Final Report: September 1981 - September 1982 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) P. Sundaram S. N. Chaudhuri		8. CONTRACT OR GRANT NUMBER(s) F08635-81-C-0235
9. PERFORMING ORGANIZATION NAME AND ADDRESS University of Tennessee Space Institute Tullahoma, Tennessee 37388		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS PE: 61102F JON: 2307-E1-20
11. CONTROLLING OFFICE NAME AND ADDRESS Air Force Armament Laboratory (DLCA) Armament Division Eglin Air Force Base, Florida 32542		12. REPORT DATE May 1983 13. NUMBER OF PAGES 150
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

Availability of this report is specified on verso of front cover.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Transonic Flow Computational Fluid Dynamics
Interference Flow-Field
Integral Equation
Viscous Interaction
Inviscid Interaction

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

An approximate method to compute the flow-field velocities around wing-body-pylon-store configurations is presented. The method adds transonic non-linear velocities to the linear velocities, computed using Hess's panel method. The non-linear effects are computed by an Integral Equation method derived for three-dimensional transonic small disturbance equations. A viscous/inviscid interaction procedure incorporates Cebeci's two-dimensional small cross flow boundary layer computations. Viscous effects over lifting

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. ABSTRACT (CONCLUDED)

and non-lifting bodies are included. The viscous effects can be simulated by either the displacement or blowing method. Also outlined is the inclusion of weak separation and wake effects in the interaction procedure. An internal flow correction is applied to predict the flow in the region between the store and the wing. A finite difference scheme to solve the full potential equations over an axisymmetric body is used to compute the transonic non-linear effects for the store-on-body pressure computation. The results obtained by the above methods are compared with the experimental results at various free stream Mach numbers and angles of attack. The results compare well at all regions except where significant interference effects are predominant, as in the case of flowfield points situated underneath the wing at 0 angle of attack and above a thick store.

PREFACE

This report was prepared by the University of Tennessee Space Institute, Tullahoma, Tennessee 37388 under Contract F08635-81-C-0235 with the Air Force Armament Laboratory, Armament Division, Eglin Air Force Base, Florida 32542. Dr. Lawrence E. Lijewski (DLCA) monitored the program for the Armament Laboratory. The work was accomplished during the period from September 1981 to September 1982.

The Public Affairs Office has reviewed this report, and it is releasable to the National Technical Information Service (NTIS), where it will be available to the general public, including foreign nationals.

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

E. C. NEWMAN, Colonel, USAF
Chief, Aeromechanics Division

Accession For	
NTIS GRA&I	☑
DTIC TAB	☐
Unannounced	☐
Justification	
PER CALL JC	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

(The reverse of this page is blank)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF FIGURES (continued)

LIST OF FIGURES (concluded)

NOMENCLATURE

a	Speed of sound
M	Mach number
p	Pressure
k	Curvature
u	Velocity in x direction or velocity in s direction
v	Velocity in y direction or velocity in n direction
w	Velocity in z direction
U	Ratio of local u velocity to the far upstream u velocity
V	Ratio of local v velocity to the far upstream u velocity
W	Ratio of local w velocity to the far upstream u velocity
UE	Experimental U velocity
UI	Internal flow corrected U velocity
UL	Linear U velocity
UT	Transonic U velocity
UV	Viscous U velocity
VE	Experimental V velocity
VL	Linear V velocity
VT	Transonic V velocity
WE	Experimental W velocity
WL	Linear W velocity
WT	Transonic W velocity
x, y, z	Coordinate directions
β	Compressibility parameter
τ	Thickness chord ratio
ψ	Stream function
ϕ	Velocity potential
ξ, η	Coordinate directions or stretching functions
δ^*	Displacement thickness
θ	Momentum thickness
ρ	Density

NOMENCLATURE (CONCLUDED)

γ	Ratio of specific heats
κ	Thermal conductivity
λ	First coefficient of viscosity
μ	Second coefficient of viscosity
H	Total pressure or metric
C_p	Pressure coefficient
r	Radius of the axisymmetric body
r_b	Radius of the body
r_3	Elliptic distance between two points
q	Dynamic pressure or resultant velocity or the produ[illegible] and δ^*
t	Time
T	Temperature
$\Delta x, \Delta y$	Step size in x and y directions

Subscripts

L	Linear values
c	Cross flow
l	Lower surface
m	Mean value or dividing point
s	Along the surface of the body
so	Source corrected
u	Upper surface
w	Wake surface
x, y, Y	Derivatives with respect to x,y,and Y variables
1	Upstream value
2	Downstream value
∞	Far upstream value

SECTION I

INTRODUCTION

The development of transonic prediction methods to complex wing-fuselage-pylon-stores combinations is of great importance because of its potential applications in fighter aircraft design, and for the detailed flow-field analysis required to ascertain the safe release of and the subsequent accurate prediction of the trajectory of the external stores after release. Several finite-difference and finite-volume computer codes developed by Jameson and Caughey (Reference 1-3) viz., FLO-27,28,and 30 can predict transonic flow including the approximation of embedded shock wave location and strength, about a swept wing attached to either a wall (plane of symmetry) or a simplified fuselage geometry. The simplified fuelage geometry varies from an infinite cylinder to a finite cylinder of moderately varying cross sections in these codes. The extension of the procedure for wing-fuselage treatment in these codes to the wing-fuselage-pylon-store case is not straightforward due to several reasons. Recent numerical experiments in this direction to investigate the direct approaches have not been very successful.

Strong viscous-inviscid interactions in transonic flow regime introduce additional complications to the exact prediction of the fluid motion. The existence of shock and the shock-induced separation regions in transonic flows is quite common. Although precise mathematical procedures for the solution of the complete set of equations describing the fluid flow are available in the literature, they are all concentrated to isolated individual areas. In the absence of a comprehensive numerical technique for a complex problem, attempts to include appropriate elaborate solutions locally, along with the global approximate solutions, have been very rewarding. Several of these ideas have been in practice in the industry. A very good example for the above is the McDonnell Douglas Aircraft Company's (MCAIR) Equivalent Simple Body (ESB) concept as discussed in Reference 4.

The validity of these approximations depend upon the significance of each of the individual areas. A first estimate of their contributions to the solution could be obtained by the asymptotic or matched asymptotic expansion principles with single or multiple

parameters representing the complexity of the problem. Earlier efforts in this area to simulate a complex flow field with Transonic Small Disturbance equations proved to be very limited (Reference 5). With the above classifications in mind, we can observe that the exact and extensive numerical computation of a certain simplified equation of motion for the transonic flow, that has significant contribution from at least more than two categories, may not yield accurate answers.

In the present report, a methodology to solve the above problems is intended. An integral equation method for the solution of the transonic small disturbance equation is applied. The method involves a non-linear correction term applied to the basic linear solution obtained with the suitable compressible coordinate transformation. The solution process as discussed in Reference 6 is also highlighted in Section II. Viscous effects are included by calculating the two-dimensional compressible boundary layer displacement thickness. A displaced body concept or Lighthill's equivalent blowing concept could be used to incorporate the viscous effects both on the non-lifting and lifting bodies. The wake effects could also be introduced in a similar way. Methods of introducing the separation effects are discussed, but the modification of the code to incorporate the separation effect is not made. A brief review of the basic scheme for viscous-inviscid interaction is outlined in Section III.

An approximate method for the flow-field velocity computation for the separated store arrangement has been developed. The experimental data for configurations 101 and 102 (see Appendix A) showed regions of shock and shock-induced separation. In the present method the shock location has been computed by a forward and backward matching procedure (in the subsonic regimes) in the solution process of two-dimensional Navier-Stokes equations. A predictor-corrector MacCormack's explicit scheme has been used to compute the two-dimensional internal flow between the wing and the store. The location and strength of the shock so computed show good agreement with the experimental values. The solution methodology along with the numerical procedure is briefed in Section IV.

Computation of the store pressure is another aspect discussed in this report. Again, the computation of on-body pressures for complex configurations is a challenging task.

The rapid decay of disturbances that could be expected at the off-body points and hence the validity of making a lower order approximations of the full Gas Dynamic equations are no longer valid for the on-body computations. But, due to the low cross flow components of the flow variables at low angles of attack, the axi-symmetric nature of the flow (assuming that the store is axi-symmetric) around the store relaxes the transonic non-linear effects appreciably. Hence, based on these experiences and due to the fact that there is no comprehensive and accurate computational procedure available for complex geometries to predict the on-body pressures, yet another semi-rigorous procedure has been developed for the store pressure calculations. The proposed procedure has been discussed in some detail in Section V.

SECTION II

INTEGRAL EQUATION METHOD

The three-dimensional Transonic Small Disturbance equation for the velocity potential, ϕ, can be written as:

$$(1 - M_\infty^2)\phi_{xx} + \phi_{yy} + \phi_{zz} = K\phi_x\phi_{xx} \tag{1}$$

where

$$K = \frac{\gamma + 1}{U_\infty} M_\infty^2$$

and the subscripts x, y, and z denote the differentiation with respect to the respective variables, γ is the ratio of specific heats, and U_∞ , M_∞ are the free stream velocity and Mach number, respectively.

Using Green's Theorem and applying Goethert's Transformation and after simplifications (Reference 6), the velocity components in the three different directions are written as:

$$\bar{u}(\bar{x}, \bar{y}, \bar{z}) = \bar{u}_L + \frac{\bar{u}^2}{2} - \frac{1}{4\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \frac{\bar{u}^2}{2}(\xi, \eta, \varsigma) \frac{\partial^2}{\partial \xi^2}\left(\frac{1}{r_3}\right) d\xi d\eta d\varsigma \tag{2}$$

$$\bar{v}(\bar{x}, \bar{y}, \bar{z}) = \bar{v}_L - \frac{1}{4\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \frac{\bar{u}^2}{2}(\xi, \eta, \varsigma) \frac{\partial^2}{\partial \xi \partial \eta}\left(\frac{1}{r_3}\right) d\xi d\eta d\varsigma \tag{3}$$

$$\bar{w}(\bar{x}, \bar{y}, \bar{z}) = \bar{w}_L - \frac{1}{4\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \frac{\bar{u}^2}{2}(\xi, \eta, \varsigma) \frac{\partial^2}{\partial \xi \partial \varsigma}\left(\frac{1}{r_3}\right) d\xi d\eta d\varsigma \tag{4}$$

where

$$\beta^2 = (1 - M_\infty^2) \quad , \quad \bar{x} = x \quad , \quad \bar{y} = \beta y \quad , \quad \bar{z} = \beta z$$

$$\bar{u} = \frac{Ku}{\beta^2} \quad ; \quad \bar{v} = \frac{Kv}{\beta^3} \quad \& \quad \bar{w} = \frac{Kw}{\beta^3}$$

$$r_3 = [(\bar{x} - \xi)^2 + (\bar{y} - \eta)^2 + (\bar{z} - \varsigma)^2]^{1/2}$$

and $\bar{u}_L$, $\bar{v}_L$, $\bar{w}_L$ are the linear velocities in the $\bar{x}$, $\bar{y}$, and $\bar{z}$ directions.

Again, after introducing a transformation and proceeding as outlined in (Reference 5)

$$\xi = Tan \quad (\frac{\pi}{2}t_1) \quad ; \quad \eta = Tan \quad (\frac{\pi}{2}t_2) \quad ; \quad \varsigma = Tan \quad (\frac{\pi}{2}t_3) \tag{5}$$

we obtain, a quadratic equation in $\bar{u}$,

$$\bar{u} = \bar{u}_L + \frac{\bar{u}^2}{2} - \frac{I_u}{2} \tag{6}$$

where I_u represents the triple integral term in equation (2).

A 6-point Gauss quadrature method is used to numerically integrate I_u. This gives the solution of the non-linear $\bar{u}$ velocity,

$$\bar{u} = 1 - \sqrt{1 - (2\bar{u}_L - I_u)} \tag{7}$$

for local subsonic flow, and

$$\bar{u} = 1 + \sqrt{1 - (2\bar{u}_L - I_u)} \tag{8}$$

for supersonic local flow.

The sonic point is determined by the vanishing of the descriminant and the smooth transition of from subsonic to supersoinc flow criterion. No predetermined position of the shock is necessary for the non-linear flow computation. The calculation of $\bar{v}$ and $\bar{w}$ velocities involves only the evaluation of the I_v and I_w integrals.

$$\bar{v}(\bar{x}, \bar{y}, \bar{z}) = \bar{v}_L - I_v \tag{9}$$

and

$$\bar{w}(\bar{x}, \bar{y}, \bar{z}) = \bar{w}_L - I_w \tag{10}$$

The computation of the linear velocities is done by the Douglas Neuman type panel method (Reference 7). The entire body is represented by linear surface source and doublet

singularities to effectively simulate the thickness and lifting effects,respectively (Figure 1). A higher order panel incorporating a parabolic vorticity distribution could also be incorporated (the present code has this capability). The inclusion of these higher order panels becomes effective only for thick wings with fairly large leading edge radii.

The panel method, although cumbersome since it requires the entire body to be represented by quadrilateral panels, has several desirable features. The present store pressure computation procedure follows the steps mentioned below:

(a) It is an integral method of solution of the Laplace Equation and hence fully conservative.

(b) Suitable for arbitrary geometry of the body.

(c) Lifting and non-lifting bodies could be handled simultaneously with as many numbers of lifting and non-lifting sections as desired.

An important point to be observed in panelling technique is that whenever there are multiply connected regions in the domain of computation, it is necessary to provide an artificial cut at an appropriate location to make the region simply-connected. Artificial cuts should be made at appropriate locations that shall not disturb the flow appreciably. This is necessitated due to the fact that the application of Green's theorem during the derivation of the integral equation to obtain the strength of source and doublet singularities requires the region to be simply-connected. This is a definite experience gained during the course of the present investigation. Examination of the integrals I_a, I_v, I_w would show that the non-linear contribution is significant only near regions of large gradients of the linear velocities, ϕ_{xx}. Hence, at the regions of rapid accelerations, the non-linear method predicts higher shock strengths compared to the physically correct values. This was observed in the cases of configurations 101 and 102. A Fortran program listing of the non-linear integral equation correction is given in Appendix B.

As the transonic parameter

$$\frac{\beta^2}{M_\infty^{4/3}\tau^{2/3}}$$

become larger, particularly when the Mach number is increased with the thickness τ being fixed (e.g., configurations 101 and 121 at $M_\infty = 0.975$), the linear velocities considerably overpredict the experimental values. This is due to excessive stretching of the co-ordinates when the compressibility corrections are made to the linear velocity. One way to overcome this difficulty is to reduce the free stream Mach number in the linear calculation. An alternative method is to introduce an artificial term in the non-linear correction procedure that will reduce this effect. An attempt in this direction was made by the principle of asymptotic expansion with a small parameter involving β. No correct estimate could be derived for various Mach numbers and thicknesses. Hence, all the linear velocity calculations were done at the same transonic Mach numbers as the experimental values. Section IV describes a local internal flow calculation procedure for fixing this problem.

It is interesting to note that in those configurations where the wing has two degrees angle of attack (configuration 201, 221, etc.) the computed results show overall good agreement with the experiment. But with the 0 degree angle of attack of the wing, regions of shock are noted. This is also shown in the shadowgraph experiments of Reference 8. The appearance of shock is due to the increased acceleretion on the bottom surface of the wing (e.g., 101, 102, etc.). At $M_{\infty'} = 1.025$, the linear solutions were obtained by using a free stream Mach number of 0.9522, which corresponds to the downstream Mach Number across a normal shock. It is indeed questionable, but the results thus obtained show fairly good agreement since the detached shock is sufficiently far upstream and is nearly normal and hence the entire panels are within the subsonic zone. At higher supersonic Mach numbers, however, these problems would be much more complicated not only because of the interaction of shocks emanating from the wing leading edge and the store nose, but also of the presence of panels that are lying outside the forward Mach cone at any point. For these problems, Woodward (Reference 9) recently suggested a third order singularity which would still allow the panel methods to be utilized.

From the above discussion, we could see that the present method of integral equation could handle fairly complicated flows within the limitations of moderately non-linear flows, particularly for off-body points velocity computations.

SECTION III

WEAK VISCOUS - INVISCID INTERACTION

3.1 Viscous Flow Modelling in Panel Methods

The simplest and most popular method of viscous correction to an inviscid flow computation is discussed in this section. In this process the matching line is not used directly. Instead, a displacement surface, deduced from the integral mass conservation property of the boundary layer replaces the real surface for the inviscid part of the calculation. This displacement surface is treated as a streamline of the inviscid flow. The implicit assumptions in this process are:

(a) The displaced surface is a real streamline.

(b) The pressure computed at the displacment surface in the inviscid calculation is the same as the pressure which drives the boundary layer.

These assumptions are valid as long as the viscous effects are weak. To link viscous and inviscid influences through displacement thickness, there are two procedures:

(i) Actually displacing the body surface by the addition of the displacement thickness locally.

(ii) Implicitly represent the displacement effect by the surface blowing or transpiration method, by imposing artifical boundary conditions on that body which simulates the displacement surface as a streamline of the inviscid flow.

Method (ii) is recommended whenever iterations are involved.

3.1.1 Displacement Thickness Method

The idea of adding the displacement thickness to the inviscid flow to incorporate the weak viscous effects could be demonstrated as follows (Reference 10).

The first order outer expansion in the asymptotic expansion of the equations of motion with the viscosity as a small parameter is :

$$\frac{\partial^2\psi_1}{\partial x^2} + \frac{\partial^2\psi_1}{\partial y^2} = B_1'(\psi_1) \tag{11}$$

where ψ_1 is stream function and B_1' is related to the vorticity .

Since the vorticity is a function of ψ_1 only, it must be constant along streamlines. For the problem of flow over a flat plate with a uniform upstream irrotational flow the solution becomes,

$$\psi_1(x,y) = Uy \tag{12}$$

which means a uniform parallel stream with a velocity U.

The first order inner solution for the semi-infinite flat plate with $Y = \sqrt{Re}y$, Re being the Reynold's number, is obtained from the equation of motion as:

$$\psi_{1YYY} + (\psi_{1_x}\frac{\partial}{\partial y} - \psi_{1Y}\frac{\partial}{\partial x})\psi_{1Y} = 0 \tag{13}$$

with the associated boundary conditions ,

$$\psi_1(x,0) = 0 \qquad \psi_{1Y}(x,0) = 0 \qquad \text{and} \qquad \psi_{1Y}(x,\infty) = U. \tag{14}$$

where the subscripts x and Y again mean derivatives with respect to the same variables.

Using one parameter Group Transformation. we can obtain the invariants of the transformation as :

$$\eta = \frac{Y}{\sqrt{2x}} \quad \text{and} \quad f_1(\eta) = \frac{\psi_1}{\sqrt{2x}} \tag{15}$$

and substituting these transformations into the differential equation (13) gives the classical Blassius boundary layer equation ,

$$f_1''' + f_1 f_1'' = 0 \tag{16}$$

with the boundary conditions ,

$$f_1(0) = f_1'(0) = 0 \quad and \quad f_1'(\infty) = U. \tag{17}$$

and the primes denote differentiation with respect to η

The second order outer expansion for the above problem is governed by

$$\frac{\partial^2 \psi_2}{\partial x^2} + \frac{\partial^2 \psi_2}{\partial y^2} = 0 \tag{18}$$

The boundary condititons near the plate are obtained by the matching condition,

$$\psi_2(x,0) = 0 \quad \text{for} \quad x < 0$$

$$\psi_2(x,0) = -Lt_{x \mapsto \infty}(Y\psi_{1Y} - \psi_1) = -\sqrt{2x}\beta \quad \text{for} \quad x > 0 \tag{19}$$

The condition far upstream is

$$\psi_2(x,y) = O(\eta) \tag{20}$$

This is the linearized problem for potential flow past the displacement parabola $Y = \beta\sqrt{\frac{2x}{Re}}$, where β =1.21678 . The second order inner expansion could be obtained as

$$\frac{\partial}{\partial Y}\left[\psi_{2YYY} + (\psi_{1_*}\frac{\partial}{\partial Y} - \psi_{1Y}\frac{\partial}{\partial x})\psi_{2Y} + (\psi_{2_*}\frac{\partial}{\partial Y} - \psi_{2Y}\frac{\partial}{\partial x})\psi_{1Y}\right] = 0 \tag{21}$$

and the associated boundary conditions ,

$$\psi_2(x,0) = \psi_{2Y}(x,0) = 0, \qquad \text{and}$$

$$\psi_{2Y}(x,Y) = 0 \quad \text{as} \quad Y \mapsto \infty \tag{22}$$

Hence , the complete second order approximation for the flow past a semi-infinite flat plate is

outer solution :

$$\psi(x,Y;Re) = UY - \frac{1}{\sqrt{Re}}\beta_1 \mathcal{R}\sqrt{2(x+iY)} \tag{23}$$

inner solution:

$$\psi(x,Y;Re) = \sqrt{\frac{2x}{Re}} \quad f_1 \quad \left[\frac{\sqrt{Re}y}{\sqrt{2x}}\right] + O(\frac{1}{Re}) \tag{24}$$

From the above explanations, we can see that the method of adding the displacement thickness by any of the two methods to account for the viscous effects in the case of wing-body configuration is a good first order approximation as long as the viscous effects are weak. Further, when the process is continued once more, i.e., obtaining the second order inner solution and deriving the third order outer solution, the contribution of this order approximation to the final solution is insignificant. Hence, we can stop the viscous/inviscid interaction calculation after one iteration to arrive at a good approximation (Reference 11 and 12). Also, Reference 11 shows that a Strip Theory approximation by solving the two-dimensional compressible boundary layer equations is adequate for the inclusion of viscous effects compared to a three-dimensional compressible boundary layer equations with small cross flow assumption.

3.1.2 Surface Blowing Method.

Since the displaced surface is assumed a streamline, the boundary condition on δ^* is that in inviscid calculations the flow direction is tangential to the displaced surfaces. With subscripts δ^* and s meaning the quantities on the displaced and on the original surfaces, respectively,

$$\frac{v_{\delta^*}}{u_{\delta^*}} = \frac{\partial \delta^*}{\partial s} \tag{25}$$

To apply transpiration method, we can write v_{δ^*} as a series expansion of the velocity on the surface with δ^* as a small parameter and with the usual definitions of u and v being the compressible velocities in the s and n directions, along and normal to the surface, respectively, i.e.,

$$v_{\delta^*} = v_s + \delta^* \frac{\partial v}{\partial n}\,|_s + 0(\delta^{*2}) \tag{26}$$

But from continuity,

$$\frac{\partial v}{\partial n} = -\frac{\partial u}{\partial s} \tag{27}$$

using (27) and (25) in (26)

$$v_s = u_{\delta^*}\frac{\partial \delta^*}{\partial s} + \delta^*\frac{\partial u}{\partial s}\,|_s \tag{28}$$

Again, writing u_{δ^*} as a series expansion,

$$u_{\delta^*} = u_s + \delta^*\frac{\partial u}{\partial n}\,|_s + 0(\delta^{*2}) \tag{29}$$

and using irrotationality criterion,

$$\frac{\partial u}{\partial n} = \frac{\partial v}{\partial s}$$

and substituting it in (29)

$$v_s = (u_s + \delta^*\frac{\partial v}{\partial s}\,|_s)\frac{\partial \delta^*}{\partial s} + \delta^*\frac{\partial u}{\partial s}\,|_s \tag{30}$$

This is the surface blowing normal velocity in terms of the velocities on the real surface which can be calculated from the same influence coefficient matrix for each iteration. The gradients can be calculated using central differencing with the δ^* values being computed using a finite difference or integral boundary layer calculation (Reference 13) procedure. As the trailing edge is approached, $\frac{\partial v}{\partial s}$ is an order of magnitude greater than u_s and hence $\delta^*\frac{\partial u}{\partial s}\,|_s$ is not negligible.

The pressure coefficient c_p is given by

$$\begin{aligned} c_p &= 1 - (q_s)^2_{\delta^*} \\ &= 1 - (u^2_{\delta^*} + v^2_{\delta^*}) \end{aligned}$$

$$c_p = 1 - (u_s + \delta^*\frac{\partial v}{\partial s}\,|_s)^2 - (v_s - \delta^*\frac{\partial v}{\partial s}\,|_s)^2 \tag{31}$$

The present scheme for viscous-inviscid interaction procedure with equivalent blowing approximation uses a three-point second order finite difference parabolic fit for the

blowing velocity

$$v_n = \frac{d}{ds}\left(u_s \delta^*\right) \tag{32}$$

Defining, $q = u_s \delta^*$,

$$\left(\frac{dq}{ds}\right)_k = \frac{q_k - q_{k-1}}{s_k - s_{k-1}}\left[1 - \frac{s_k - s_{k-1}}{s_{k+1} - s_{k-1}}\right] + \frac{q_{k+1} - q_k}{s_{k+1} - s_k}\left[1 - \frac{s_{k+1} - s_k}{s_{k+1} - s_{k-1}}\right]$$

$$\text{for} \qquad k = 1, 2, \cdots, (N-1) \tag{33}$$

and

$$\left(\frac{dq}{ds}\right)_N = \frac{(s_N - s_{N+1})q_{N-2}}{(s_{N-1} - s_{N-2})(s_N - s_{N-2})} - \frac{s_N - s_{N-2}}{(s_{N-1} - s_{N-2})(s_N - s_{N-1})} q_{N-1}$$

$$+ \frac{2s_N - s_{N-1} - s_{N-2}}{(s_N - s_{N-2})(s_N - s_{N-1})} q_N \tag{34}$$

for the last panel.

3.1.3 Kutta Condition

Inviscid Flow

Explicitly:

(a) Equate the upper and lower surface velocities at or near the trailing edge.

(b) Specify a velocity component usually (but not necessarily zero) normal to the extended camber in the wake.

Implicity:

(a) The basic formulation of the mathematical model inherently includes a condition which yields a smooth flow at the trailing edge.

Viscous flow

The rule of stagnation pressures (obtained from potential flow) at the non-cusped trailing edge to control circulation is not correct. But from boundary layer assumption

and the matching line assumption, we can imply that the pressure outside the displacement surface is the same as that on the body. Hence, the condition to control circulation is that pressures are equal on the displacement surface on both sides at the trailing edge. This criterion transforms in the case of the surface blowing case as

$$u^2_{\delta^*_u} + v^2_{\delta^*_u} = u^2_{\delta^*_\ell} + v^2_{\delta^*_\ell} \tag{35}$$

$$u_{\delta^*_u} - u_{\delta^*_\ell} = \frac{\left(v^2_{\delta^*_l} - v^2_{\delta^*_u}\right)}{\left(v_{\delta^*_\ell} + u_{\delta^*_u}\right)} \tag{36}$$

which is similar to the inviscid case of equal velocities when $u_{\delta^*_u} = u_{\delta^*_\ell}$. Equation (36) can be written as:

$$u_{\delta^*_u} = u_{\delta^*_l} + \left(v_{s_l} - \delta^*_l \frac{\partial u}{\partial s}\,|_{s_\ell}\right)^2 - \frac{\left(v_{s_u} - \delta^*_u \frac{\partial u}{\partial s}\,|_{s_u}\right)^2}{\left(u_{s_u} + \delta^*_u \frac{\partial v}{\partial s}\,|_{s_u}\right) + \left(u_{s_l} + \delta^*_\ell \frac{\partial v}{\partial s}\,|_{s_\ell}\right)}$$

$$-\delta^*_u \frac{\partial v}{\partial s}\,|_{s_u} - \delta^*_\ell \frac{\partial v}{\partial s}\,|_{s_\ell} \tag{37}$$

Note that all the terms containing δ^* are on the RHS and consequently the influence coefficient matrix remains unaltered. The subscripts u and l for δ^* and s denote the upper and lower surfaces for the last panel near the trailing edge. No special treatment has been made for the Kutta condition except that the usual condition of equal pressures on the centroid of the last panel on the top and bottom displaced surfaces is imposed.

The boundary layer equations are solved by the Keller's Box Method (Reference 13 and 14). The flow quantities are solved at the center of each mesh with the pressure gradient obtained from the external inviscid velocities. We note that the displacement thickness is the only boundary layer parameter that affects the modified potential flow. Special care has been taken to determine the stagnation point of each lifting section. *Computationally, this location is found by taking the dot product of the velocity at the* mid-points of each panel with the average of the unit tangent vectors of the two N-lines bordering the panel. This dot product changes sign exactly at one point in each lifting

strip. The desired stagnation point is obtained by interpolation with respect to values of the dot product between the mid-points on either side of the sign change. The boundary layer calculation can be carried out for both laminar and turbulent cases. The turbulent boundary layer calculations are done using a mixing length model. The calculations predict the separation point . The present code can automatically incorporate any of the above mentioned two viscous effects at only one angle of attack at a time. compared to the possibility of calculating the potential flow for up to ten angles of attack.

For small angles of attack and thin wings the boundary layer effects are small. A further modification of the code to incoporate the boundary layer computation on axi-symmetric body is effected. This requires the non-lifting body panel M-lines and N-lines to be interchanged in direction compared to the potential part. The panels are redistributed from tip to nose on the bottom surface followed by nose to tip on the top surface. The necessary modifications in the panel surface normal velocity vector calculations and the velocity vector computations are made. No special treatment for stagnation point determination has been introduced, since most non-lifting bodies under study have a fairly pointed nose shape. Another justification for this is the fact that a slight deviation of the stagnation point location is not expected to alter the store flow field appreciably. A Fortran listing of all these modifications is included in Appendix C.

3.1.4 WAKE EFFECTS

The presence of the wake has two distinct contributions on the wing. They are essentially

(i) Displacement effect.

(ii) Curvature or pressure discontinuity effect.

The displacement effect of the wake is precisely analogous to that of the boundary layer over the solid surface of the wing. Thus it can be represented either by

(a) source distribution over an infinitesimally thin surface. leading to a discontinuity in velocity normal to the surface given by,

$$\Delta v_{n_w} = \frac{d}{ds}\left(u_{s_w}\delta^*_w\right) \qquad \text{or} \tag{38}$$

(b) as a streamtube of thickness δ^*_w.

where the subscript w refers to the conditions on the wake surface.

In either case the normal first-order definition of δ^*_w will suffice, since the effect of the wake on the pressure distribution of the wing is small.

To compute δ^*_w on the wake using the panel method, the following procedure might be used:

(1) During the inviscid computation process, distribute off-body points close to the approximate wake surface and obtain the inviscid velocites at those points.

(2) In the boundary layer calculations, modify the airfoil shape locally to include this wake surface and hence obtain the δ^*_w.

For an inviscid flow, and approximately in viscous flow (Figure 2),

$$\frac{\partial p}{\partial n} = k\rho u^2 \tag{39}$$

where k is curvature of the streamline and ρ is the density. Integrating across the wake, we have

$$p_u - p_l = -\int k\rho u^2 dn$$

$$p_u - p_l = -k\rho_w u^2_{s_w}\left[\delta_w - \left(\theta_w + \delta^*_w\right)\right] \tag{40}$$

where δ_w and θ_w are the boundary layer and momentum thicknesses, respectively. Again, considering the pressure jump across the wake in the sense of source distribution,

$$\begin{aligned} \Delta p &= (p_u - p_l) + k\delta_w u^2_{s_w}\delta_w \\ \Delta p &= k\rho u^2_{s_w}\left(\delta^*_w + \theta_w\right) \end{aligned} \tag{41}$$

since Δp is small, it could be written as

$$\Delta u = -ku^2_{s_w}\left(\delta^*_w + \theta_w\right)$$

The effect described above is similar to that of a Jet flap, but of opposite sign, with the jet momentum coefficient replaced by $\left(\delta^*_w + \theta_w\right)$.

The determination of the exact wake shape is fairly complex and hence within the present order of approximations, we can make certain simplifying assumptions. An effective way to determine the shape of the wake is to require that the wake surface originates from the trailing edge and that its tangent bisects the total velocity vectors on the upper and lower side of the wake surface i.e.,

$$\left(\frac{v_{n_w}}{u_{s_w}}\right)_u = \left(\frac{v_{n_w}}{u_{s_w}}\right)_\ell \tag{42}$$

In agreement with the boundary layer theory, for the determination of the wake shape alone we impose zero pressure jump across the wake (curvature effect is negligible), which implies

$$(v_{n_w})_u^2 + (u_{s_w})_u^2 = (v_{n_w})_\ell^2 + (u_{s_w})_u^2 \tag{43}$$

from which follows that

$$(v_{n_w})_u = (v_{n_w})_\ell \quad \text{and} \quad (u_{s_w})_u = (u_{s_w})_\ell \quad \text{on the wake} \tag{44}$$

Thus, the tangential velocity is continuous across the wake and the normal velocity is symmetric, and thus discontinuous, with respect to the wake. Thus, the wake effect could be represented by a source distribution on the wake only.

For non-lifting bodies, the effect of wake is necessary to be included. The wake of the non-lifting blunt based body is quite complex because of the base flow. But within the order of approximations incorporated in the other parts, the wake of a non-lifting axisymmetric body could be represented as an effective body of equivalent base shape for inviscid calculations. The shape of the equivalent base could be approximated as a backward facing step in two dimensions. Previous experiments conducted in UTSI (Reference 15) have yielded that inviscid streamline for the backward facing step reattached the sting behind the base of the step at approximately six base heights from the base. This relation was utilized when repanelling the non-lifting bodies for the inclusion of viscous effects.

3.1.5 Separation Effects

Inclusion of weak separation effects in an overall sense is another step toward improving an airfoil or wing viscous-inviscid interaction computational scheme. The separation effects could be due to strong or weak viscous inviscid interactions. The flow separation due to strong interactions like shock boundary layer interaction is very complex to incorporate. The weak interaction induced separation could be successfully modeled in the following way.

The near flow field of the airfoil is divided into four different regimes (Figure 3). The conventional inviscid panel methods could be applied at Region 1. For Region 2, the integral or differential boundary layer equation approximations are adapted. For the free shear layers, Region 3, vortex sheets of constant density along their length starting from the upper and lower separation points (the lower surface separation point being taken as the trailing edge for positive angle of attack). The shape of the vortex sheet is not known a priori, and it is iteratively determined as a streamline of the resultant flow. The vortex strength on the upper and lower surface sheets are set to be equal. Thus, the modified airfoil shape is determined for the next inviscid computations. In Region 4, the reduced total head must be taken into account when calculating the body pressures at points downstream of separation. The two basic assumptions embedded in this categorization of the regions are:

(1) The boundary layer and the free shear layers are thin and hence could be represented as slip surfaces which are streamlines across which there exists a jump in velocity.

(2) The wake does not have significant vorticity and has constant total pressure (lower than the free stream total pressure) and hence a potential flow region.

The usual boundary conditions of $\underline{V}.\underline{n} = 0$ for potential flow and

$$v_n = \frac{\partial}{\partial s}\left(u_s \delta^*\right)$$

for viscous-inviscid interaction are applied.

The following approximate relations are empirically determined by earlier investigators (Reference 16) and could be judiciously used (see Figure 3 for symbols)

$$W_L = W_H + 1.113 + \frac{0.192}{(t/c)} - \left(\frac{0.72}{t/c}\right)^2 \tag{45}$$

For the upper shear layer,

$$V_m = \frac{1}{2}(V_{outer} + V_{inner}) \qquad = \qquad \text{average velocity in the layer} \tag{46}$$

where,

$$V_{outer} = V_m + \frac{1}{2}\gamma_u$$

$$V_{inner} = V_m - \frac{1}{2}\gamma_u \qquad \text{and}$$

$$\gamma_u = V_{outer} - V_{inner}$$

is the vorticity in the upper shear layer (and vice versa for the lower). The zero static pressure drop across the shear layer implies that the total pressure jump across the shear layer could be calculated as

$$\Delta H = p_{inner} + \frac{1}{2}\rho\left(V_m - \frac{1}{2}\gamma_u\right)^2 - \left[p_{outer} + \frac{1}{2}\rho\left(V_m + \frac{1}{2}\gamma_u\right)^2\right] \tag{47}$$

with

$$\Delta H = H_{inner} - H_{outer}$$

Since the wake is assumed to have constant total pressure, the jump in total pressure across the free shear layer is the same everywhere in the wake.

The modified pressure coefficient is given by

$$C_p = 1 - \left(\frac{V}{V_\infty}\right)^2 + \frac{\Delta H}{q_\infty} \tag{48}$$

where V_∞ and $q\infty$ are the free stream velocity and dynamic pressures, respectively. The above discussion is appropriate only for weak separation which occurs at the trailing edge of wing at moderate angles of attack. Strong separation induced by shock boundary layer interactions and by high angles of attack are more complex (Reference 17).

SECTION IV

INTERNAL FLOW COMPUTATION

The results obtained with the method of flow-field velocity computation, as discussed in Section II, were in reasonable agreement with the experimental data for most configurations excepting those which included field points in the gap region that appears between the wing and the store (Configurations 101 and 102). These configurations essentially simulated stores shortly after separation. The transonic flow in these regions should contain shocks, as observed in experimental data, and strong viscous inviscid interactions. The shocks in these regions were smeared off in the linear theory, and the non-linear equation method predicted wrong location and strength of the shocks. This region should have to be analyzed by a much more elaborate method with the representation of the physics of this complex flow through an adequate mathematical modelling. Hence, in the hierarchy of the approximate analysis of the flow field, this gap region has been investigated independently by a finite difference method for solving the two-dimensional full Navier-Stokes equations (the two-dimensional section being the vertical plane passing through the store centerline where the experimental data was available). The Navier-Stokes equations were written in characteristics form. MacCormack's explicit, time marching, factored, predictor-corrector, finite difference scheme (Reference 18) has been utilized to compute the internal flow between the two walls (underneath the wing and above the store). Because of the inherent resistance of the streamline to change their curvature at transonic flow, the two-dimensional approximation is relevant. The approximate inlet and exit boundary conditions at the appropriate mesh points are obtained from the external flow computations, treating those points as off-body points.

The two-dimensional Navier Stoke's equations are written as:

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x}(\rho u) + \frac{\partial}{\partial y}(\rho v) = 0$$

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + \frac{1}{\rho}\frac{\partial p}{\partial x} = \frac{1}{\rho}\frac{\partial}{\partial x}\left[(\lambda + 2\mu)\frac{\partial u}{\partial x} + \lambda\frac{\partial v}{\partial y}\right] + \frac{1}{\rho}\frac{\partial}{\partial y}\left[\mu\left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}\right)\right]$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + \frac{1}{\rho}\frac{\partial p}{\partial y} = \frac{1}{\rho}\frac{\partial}{\partial y}\left[(\lambda + 2\mu)\frac{\partial v}{\partial y} + \lambda\frac{\partial u}{\partial x}\right] + \frac{1}{\rho}\frac{\partial}{\partial x}\left[\mu\left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}\right)\right]$$

$$\frac{\partial \rho}{\partial t} + u\frac{\partial p}{\partial x} + v\frac{\partial p}{\partial x} + v\frac{\partial p}{\partial y} = a^2\left[\frac{\partial \rho}{\partial t} + u\frac{\partial \rho}{\partial x} + v\frac{\partial \rho}{\partial y}\right] + (\gamma - 1)\Big[(\lambda + 2\mu)(\frac{\partial u}{\partial x})^2$$

$$+(\lambda + 2\mu)(\frac{\partial v}{\partial y})^2 + \mu(\frac{\partial v}{\partial x})^2 + \mu(\frac{\partial u}{\partial y})^2$$

$$+2\lambda\frac{\partial u}{\partial x}\frac{\partial v}{\partial y} + 2\mu\frac{\partial u}{\partial y}\frac{\partial v}{\partial x} + \frac{\partial}{\partial x}(\kappa\frac{\partial T}{\partial x}) + \frac{\partial}{\partial y}(\kappa\frac{\partial T}{\partial y})\Big]$$

where ρ is the density, p is the pressure, T is the temperature, u and v are the velocity components in x and y directions, a is the speed of sound, R is the gas constant, μ and λ are the first and second coefficients of viscosity, respectively; γ is the ratio of specific heats, κ is the thermal conductivity, x and y are the space coordinates, t is the time.

The equations are solved using a predictor-corrector explicit MacCormack's scheme. The predictor step uses a backward differencing and the corrector step uses a forward differencing to the convection terms. Central differencing is used for the diffusion terms in both predictor and corrector steps. The scheme is stable provided that the time step increment satisfies the stability criterion,

$$\Delta t = \frac{A}{\left[\frac{|u|}{\Delta x} + \frac{|v|}{\Delta y} + a\sqrt{\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2}}\right]}$$

where A is a parameter like Courant Fredrick Levy (CFL) number which neeeds to be set less than 1 for the cases of supersonic calculations with shock. A fourth order artificial viscosity and second order artificial thermal conductivity terms are added to dampen the numerical oscillations near shocks. The time split procedure allows the marching in two different directions independently by introducing a term which is of the same order as the truncation error. The steady state solutions are obtained by the asymptotic unsteady calculations.

The solution procedure automatically computes the initial two-dimensional flow surface by assuming one-dimensional approximation based on the pressure, velocity, and density values specified at the inlet of the convergent divergent section. The boundary conditions of impermeable and either slip or no-slip walls, together with the exit pressure

conditions, are satisfied to obtain the final solution surface. The truly external flow calculation would yield maximum velocities at those regions which are closest, i.e., at the minimum area regions, whereas, the internal flow calculation would have the throat (minimum area) choaked. The pressure distribution upstream readjusts itself so as to yield sonic velocity near the throat. There may be a further acceleration to supersonic speed downstream and finally recover the exit pressure through a near normal shock. For the present problem involving the flow in the gap between the pylon or wing and the store is not essentially two-dimensional or completely internal flow.

The basic VNAP (Viscous Nozzle Analysis Program) scheme described above does pose difficulties to capture shock at transonic speeds. The scheme was essentially developed for fully subsonic or fully supersonic and plug nozzle flows. The capturing of shocks should be done by carefully adjusting the artificial viscosity. Originally, the computed flow either continued to accelerate to supersonic speeds or smoothly decelerated subsonically downstream of the sonic points for the cases of supersonic or subsonic exit boundary conditions, respectively. In order to modify the scheme to obtain a solution with shock, an indirect approach to fit the shock by a forward and backward sweep process has been adapted. During the forward march normal shock was assumed to be present at all supersonic stations and the Rankin-Hugoniot jump relations have been imposed at all those stations. The downstream Mach numbers across the normal shock, M_2^+, at each supersonic station is then computed. A reverse sweep from the exit station with the approximate switched inlet boundary conditions is then made with the modified wall and centerbody shapes. These modified shapes are obtained by adding the displacement thicknesses (available from the previous forward sweep) to the original shapes. The points where the M_2^+ matches with the Mach number obtained during the backward sweep M_1^- determines the shock location. Figures 4, 5, and 6 show the above idea pictorially. The shock locations determined from the procedure discussed above compare well with those observed in the experimental results for both the configurations. Figure 7 and 8 show the direct computation of the shock location for the above two cases by carefully adjusting the artificial viscosity terms. The same values of the artificial viscosity terms are used for the other configuration internal flow computations as well.

SECTION V

COMPUTATION OF STORE PRESSURE DISTRIBUTION

The store pressure distribution computation procedure described herein incorporates the attractive features of the different methods of solution of the transonic multibody problem. The flexibility of the panel code for complex geometries and the accuracies of the finite difference codes are fused together to yield an approximate method which does not involve very elaborate computations. The formulation of the problem and the solution procedure for store pressure calculation by the RAXBOD (Relaxation solution of AXi-symmetric BODy) finite difference scheme is outlined below (Reference 19).

The exact equation for the disturbance potential, ϕ, for axi-symmetric compressible flow can be written in orthogonal curvilinear coordinates ξ, η as:

$$\left(1-\frac{u^2}{a^2}\right)\frac{1}{H}\frac{\partial}{\partial\xi}\left(\frac{1}{H}\frac{\partial\phi}{\partial\xi}\right)-2\frac{uv}{a^2H}\frac{\partial^2\phi}{\partial\xi\partial\eta}+\left(1-\frac{v^2}{a^2}\right)\frac{\partial^2\phi}{\partial\eta^2}+\left[\frac{k}{H}\left(1-\frac{u^2}{a^2}\right)+\frac{\cos\theta}{r}\right]\frac{\partial\phi}{\partial\eta}$$
$$+\left[2\frac{kuv}{Ha^2}+\frac{\sin\varphi}{r}\right]\frac{1}{H}\frac{\partial\phi}{\partial\xi}=0 \qquad (48)$$

where θ and k are the angle (measured counterclockwise from the axis of symmetry) and curvature of the reference coordinate surface, respectively, $H = 1 + k\eta$ is the metric, a is the local speed of sound, u and v are the ξ, η components of the total velocity, q, and r is the radius from the axis.

$$\frac{\partial\phi}{\partial\xi} = H(u - \cos\theta)$$
$$\frac{\partial\phi}{\partial\eta} = v + \sin\theta$$

The coordinate system is built in a special way so as to have $\eta = 0$ on the body surface throughout and to remove the difficulties associated with the fully body fitted coordinate system in the concave part of the body surfaces. Hence, in the forebody region, the body normal coordinates are used upto the first horizontal tangent and beyond that point a sheared cylindrical system has been introduced.

A convenient normal coordinate stretching function has been applied. It is represented as

$$\eta = \frac{Ay}{(1-y)^{\alpha}} \tag{49}$$

where η is the physical coordinate normal to the body and y is the computational coordinate which varies from zero at the body surface to one at infinity. The constant A could be used to control the physical step size at the body, $A = (\frac{d\eta}{dy})_{y=0}$ and for a given value of A, the exponent α is used to control the size of the last finite value of η (the larger α is, the farther the points are away from the body). The tangential coordinate stretching is based on the physical arc length along the reference surface.

$$\xi = \frac{\xi_{\max}}{2} + (x - \frac{1}{2})\left[A + B(x - \frac{1}{2})^2\right] \tag{50}$$

where A and B are determined by specifying $\left(\frac{d\xi}{dx}\right)_{x=0}$ and requiring that $\xi = \xi_{\max}$ at $x = 1$. Hence,

$$A = \frac{3\xi_{\max} - \left(\frac{d\xi}{dx}\right)_{x=0}}{2} \tag{51}$$

and

$$B = 4(\xi_{max} - A) \tag{52}$$

For open bodies the tangential coordinate stretching is divided into two regions with the physical location of the dividing point, x_m. The stretching function used for the region from the nose up to x_m is

$$\xi = \lambda_1 x + \lambda_2 x^3 + \lambda_3 x^5 + \lambda_4 x^7, \quad 0 \leq x \leq x_m \tag{53}$$

and in the region from x_m to unity, the stretching function

$$\xi = \xi_m + b\frac{(x - x_m)(1 - x_m)}{(1 - x)}, \quad x_m \leq x < 1 \tag{54}$$

is applied. The coefficients in these expressions are determined by specifying ξ_m, $(\frac{d\xi}{dx})_{x=0}$, and $\frac{d^2\xi}{dx^2}$ be continuous at $x = x_m$.

In the sheared cylindrical coordinates, the potential equation is

$$\left(1 - \frac{u^2}{a^2}\right)\frac{\partial^2\phi}{\partial\xi^2} - 2\left[\frac{\partial r_b}{\partial x}\left(1 - \frac{u^2}{a^2}\right) + \frac{uv}{a^2}\right]\frac{\partial^2\phi}{\partial\xi\partial\eta}$$

$$+\left[\left(1-\frac{v^2}{a^2}\right)+\left(\frac{\partial r_b}{\partial x}\right)^2\left(1-\frac{u^2}{a^2}\right)+2\frac{uv}{a^2}\frac{\partial r_b}{\partial x}\right]\frac{\partial^2\phi}{\partial\eta^2}$$

$$+\left[-\frac{\partial^2 r_b}{\partial x^2}(1-\frac{u^2}{a^2})+\frac{1}{r}\right]\frac{\partial\phi}{\partial\eta}=0 \tag{55}$$

where ξ is again identified with the axial coordinate x, and η is a transformed radial coordinate such that $\eta = 0$ on the body surface, and r_b is the local body radius. In the sheared cylindrical system,

$$\frac{\partial\varphi}{\partial\xi}=u+\frac{\partial r_b}{\partial x}v-1, \quad \text{and} \quad \frac{\partial\phi}{\partial\eta}=v \tag{56}$$

The boundary conditions are

$\underline{\eta\to\infty}$

$$\phi\to 0 \tag{57}$$

$\underline{\eta=0}$

$$v=0, \quad \text{or} \quad \frac{\partial\phi}{\partial\eta}=\sin\theta \quad \text{for} \quad x\le x_m \tag{58}$$

and

$$v-u\frac{\partial r_b}{\partial u}=0 \ \text{ or } \ \frac{\partial\phi}{\partial\eta}=\frac{\frac{\partial r_b}{\partial u}}{\left(1+\frac{\partial r_b}{\partial u}\right)}\left(1+\frac{\partial\phi}{\partial\xi}\right) \ \text{ for } \ x\ge x_m \tag{59}$$

The computational scheme incorporates a rotated differencing at supersonic points, where $u^2, v^2 < a^2 < u^2 + v^2$, to avoid the possible difficulties that would arise during the solution process because of the misalignment of the sonic line with the η coordinate surface. The rotated differencing scheme replaces the highest derivative part of the differential equation (48) from ξ, η coordinates to s, n coordinates which are nearly along and normal to the velocity vector direction locally by

$$\left(1-\frac{q^2}{a^2}\right)\phi_{ss}+\phi_{nn} \ =0 \tag{60}$$

where

$$\phi_{ss} = \frac{1}{q^2}\left[\frac{v^2}{H}\frac{\partial}{\partial\xi}\left(\frac{1}{H}\frac{\partial\phi}{\partial\xi}\right) + 2\frac{uv}{H}\frac{\partial^2\phi}{\partial\xi\partial\eta} + v^2\frac{\partial^2\phi}{\partial\eta^2}\right]$$

and

$$\phi_{nn} = \frac{1}{q^2}\left[\frac{v^2}{H}\frac{\partial}{\partial\xi}\left(\frac{1}{H}\frac{\partial\phi}{\partial\xi}\right) - 2\frac{uv}{H}\frac{\partial\phi}{\partial\eta} + u^2\frac{\partial^2\phi}{\partial\eta^2}\right]$$

for the body-normal coordinates and

$$\phi_{ss} = \frac{1}{q^2}\left[u^2\frac{\partial^2\phi}{\partial\xi^2} + 2u\left(v - \frac{\partial r_b}{\partial x}u\right)\frac{\partial^2\phi}{\partial\xi\partial\eta} + \left(v - u\frac{\partial r_b}{\partial x}\right)^2\frac{\partial^2\phi}{\partial\eta^2}\right]$$

and

$$\phi_{nn} = \frac{1}{q^2}\left[v^2\frac{\partial^2\phi}{\partial\xi^2} - 2v\left(u + \frac{\partial r_b}{\partial x}v\right)\frac{\partial^2\phi}{\partial\xi\partial\eta} + \left(u - v\frac{\partial r_b}{\partial x}\right)^2\frac{\partial^2\phi}{\partial\eta^2}\right]$$

Further details of the above procedure could be obtained from Reference 20. The code incorporates a successive mesh refinement from a coarse to a fine mesh as many sequential steps as desired within the purview of storage space and CPU time limitations.

The present store pressure computation procedure follows the steps mentioned below:

(1) Compute the store only and the entire configuration on-store pressure distributions using the panel method.

(2) Calculate the interference effects by subtracting the store alone on-body pressures locally from that of the entire configuration store on-body pressures.

(3) Compute the store only on-store pressure distribution using the RAXBOD finite difference code. Note here that a linear or higher order interpolation in velocity is needed to find the pressures at the same points as those of the panel method.

(4) Add the pressure coefficient due to interference and the store alone on-body finite difference computations to yield a corrected on-store pressure.

The above procedure is questionable in the sense that the linear combination of the flow quantities is attempted in a non-linear flow (both due to geometry and due to the equations of fluid motion). A more reasonable approach would be to build in correlations between the store alone and wing-body-store combination on store pressures and between

the panel and finite difference methods at the same transonic Mach numbers for the store alone case. Several such efforts have been tried, and at the time of reporting a comprehensive correlation was not arrived at. In retrospect, the results of the procedure discussed formerly have only been presented. It could be seen that the above scheme did not improve panel method prediciton very much. A much more elaborate way of incorporating the finite volume codes (FLO-27) for wing-body configurations and panel method for the entire flow field is being investigated, and it is speculated to yield improved predictions.

SECTION VI

RESULTS AND DISCUSSION

In this section the results of the calculation procedures discussed in the previous sections for different configurations are presented. The present study has been mainly directed towards improving the results obtained in the previous approaches and arriving at a unified and simple procedure for the analysis of complex wing-body-pylon-store configurations at transonic speeds.

The experimental data base for the present theoretical comparison comprises the data obtained from the Arnold Engineering Development Center (AEDC) 1foot x 1foot transonic wind tunnel (1T) Laser Velocimeter flow-field survey experiments for configurations (a) shown in Appendix A, and from the AEDC 4foot x 4foot transonic wind tunnel (4T) 5 hole pressure probe flow-field survey for Configuration 24 of configurations (b) given in Appendix A. The configurations (a) have wall-mounted 45-degree swept wing with Aspect Ratio (AR) of 2.75 having NACA 0005-34 airfoil sections mounted with two different store shapes. The free stream Mach numbers of test were 0.92, 0.975, and 1.025 with wing angles of attack of 0 degree and 2 degrees at Reynolds number of about 3 million per foot. Configuration 001 is similar to Configuration 221, but with the inclusion of a fuselage-mounted wing instead of the wall-mounted wing. Configuration 001 has been tested at a subcritical Mach number 0.7. Configurations 24 and 13 contained in configurations (b) have 40-degree swept wings with AR=2.1 and 4 percent thick circular airfoil sections. Configuration 13 is similar to Configuration 24 excepting that it has a pressure instrumented store instead of a dummy store as in Configuration 24. Configuration 24 has been tested at the free stream Mach numbers of 0.925 and 0.950 at 0 degree, 2 degrees, and 5 degrees angles of attack.The set of experiments on Configuration 13 to form a data base for the store pressure distribution computation was also conducted at the AEDC 4T wind tunnel at free stream transonic Mach numbers of 0.925, 0.952, and 1.052 at 0 degree, 2 degrees and 5 degrees angles of attack.

In the earlier analysis we obtained good agreement between theory and experiment

for simple wing-body and wing-body-pylon configurations as could be observed from Figures 9 to 14. The additional complexities like wing at 0 degree incidence, thus increasing the flow accelerations below the wing which is the region of interest as compared to the positive angles of attack of the wing cases, and the separated store in the close proximity of the wing and/or pylon involve considerable interference effects. Some of these are first order and some others are second order. In some regions of the flow-field first order theory is adequate, whereas second order corrections are required for some other portions. In the process of present investigation, we identify these regimes and indicate the ways of improving the results further.

AFFDL configuration-24 with wing-fuselage-pylon and store showed considerable deviation from the experiments in the previous analysis. The introduction of additional panels along with their better distribution improved the solution for the flow-field velocities upstream of the wing trailing edge. The introduction of non-linear effects improved the results slightly (Figures 15 to 19). Also, in this case the equivalent blowing method to account for weak viscous-inviscid interaction outlined in Section III has yielded the results as presented in Figure 20. The strong viscous interaction regions near the wing trailing edges as seen in Figure 20 are beyond the scope of the present first order theories and hence the sudden accelerations aft of x=18 and the subsequent deceleration (Figures 14 to 19), typical of the near wake regions, are not predicted well. The thin wing shapes and the low angles of attacks involved in the present configurations introduce only minor corrections to the off body points velocities as implied in equation (29) because of the rapid decay of the tangential velocity in the direction normal to the surface of the wing. Also, a lower Mach number of 0.82 was used in the linear computation instead of the actual free-stream value of 0.925 to examine the influence of local compressibility coordinate stretching on the flow-field velocities for linear calculations. These results are shown in Figure 21. The results show negligible improvements in the predicted flow-field velocities. Hence, the same free stream Mach number as the wind tunnel test value has been used in all the free stream subsonic linear computations. The transonic non-linear corrections introduced by the integral equation method as described in Section II have been incorporated on the solution obtained after the viscous-inviscid interaction scheme. Thus, we can observe that the inclusion of weak viscous-inviscid interaction together

with nonlinear corrections improve the overall prediction, but the near wake regions could not be adequately simulated with the present scheme as expected.

It is well proven that the panel method gives good results for subcritical flow over arbitrary three-dimensional geometries. To test this idea again for the present case of separated store configuration a trial case, configuration 001, was included in the study at a free stream subsonic Mach number of 0.7. Figures 22 to 24 show the results of this calculation. The results show good agreement (Figures 22 and 23) except in the gap region between the pylon and the store. The two-dimensional internal flow calculation as described in Section IV has been subsequently incorporated for the flow in this gap region and these results show considerable improvement as shown in Figure 24. In the absence of a better correlation the internal and external effects have been averaged for the present cases.

The next configuration studied incorporated the store separation effects on the flow-field. This configuration has the wing at 2 degrees angle of attack and hence the flow-field underneath the wing does not experience considerable supercritical accelerations. Figures 25 and 26 show the U velocity distribution along the store axis for various positive z locations. Since the likely internal flow in the gap does not involve appreciable choking effects, the external flow computation alone yielded good predictions. The shock location and its strength were not determined exactly. Good predictions have been observed at the points below the store as seen in Figure 27. The viscous effects were considered insignificant for this case and hence have not been implemented. A similar study to understand the influence of lowering the Mach number below M_{cr} for linear calculations yielded slightly different solutions as shown in Figures 28.

The earlier computation results on configuration 221 which showed considerable deviation between theory and experiment in the positive z values were identified to be due to the presence of multiply connected regions. This error was fixed by introducing an artificial cut connecting the wind tunnel wall and the sting store junction. The results presented in Figures 29 and 30 show significant improvement. The artificial cut near the pylon trailing edge has introduced some spurious acceleration aft of x=3.7. The linear and non-linear results did not show much deviation and no shock was observed in the

flow.

The effects of the store are more significant in the regions between the store axis and the wing for the configurations without pylon due to the pressure gradient imposed by the wing bottom surface in addition to that of the store shape. Hence, the configurations 101 and 102 show regions of shock and shock boundary layer interaction. In addition, the internal flow between the wing and the store becomes an important factor for the flow-field velocities near the store for positive z values. The overall prediction for Mach number of 0.92 is fairly good at all z values as could be observed from Figures 31 to 36. Two shocks, one near x=2.25 and the other near x=4.0, are closely predicted. Detailed comparisons of the theoretical and experimental V and W velocities are avoided in most of the configurations because of the uncertainities of those data due to experimental intricacies. When the Mach number was increased to 0.975, the flow changed its behavior (Figures 37,38 and 47). The stronger shock that occurs near x=4 makes the flow fully choked and hence predominantly internal. The shock that was observed near x=2.25 for M=0.92 has disappeared in this case. The region of shock-induced separation downstream of x=3.9 is evident from the experimental data given in Figure 47 from the slow recovery of the pressure downstream. Although the present calculations show the shock near x=4, the pressure recovery downstream is fairly rapid and hence the discrepancy between the theory and experiment.

The influence of the internal flow has been felt in the inboard and outboard sides of the store as well. In the absence of any correlation between the correction at the store centerline and at other regions above the store, no correction for the internal flow computation has been applied at these locations. Since the mass flow readjusts itself so that the throat becomes choked, the predicted velocities by external singularity methods are higher than the data as shown in Figures 37 and 38. The source correction method used by other investigators to simulate the flow fields near the engine nacelle could be utilized to correct the inboard and outboard stations. Similar technique to the one used for configuration 101 has been applied to configuration 102. A thicker store in this case introduces a stronger shock. The shock location was upstream of the actual location and its strength has been overpredicted as seen in Figure 39.

Figures 40 shows the results of the introduction of pylon to the previous configuration along a section outboard of the axis. The non-linear calculations do not show the shock near the shoulder of the store which might be due to the fact there is no sharp gradient of the linear velocities upstream. Physically, the data shows a separation bubble on the store downstream of x=2.8 and hence there is an acceleration aft of x=3.0. Incipient separation downstream of x=3.25 is observed. These details could not be predicted by the present scheme and require elaborate computations near this section.

Next, the results of configuration 121 at Mach numbers 0.92, 0.975, and 1.025 are shown in Figures 41 to 46. For supersonic free stream, consistent with what has been observed in the shadowgraph pictures in Reference 8 and in the computed results from FLO27 code, a detached near normal shock was assumed upstream of the store leading edge and the adiabatic shock relation there yielded a Mach number downstream of the shock to be 0.9522. The non-linear calculations carried out at this Mach number did not show any shock or significant viscous effects. The results show reasonable agreement with the experimental data.

The forward and reverse sweep technique discussed in Section IV has next been applied to configurations 101 and 102 to locate the shocks properly. Figures 47 and 48 show the results after this computation. The results obtained without the addition of the displacement thickness during the reverse sweep did not show correct shock location for Configuration 102. The addition of the displacement thickness to either walls indeed predicted the shock location fairly accurately for both the stores. Subsequent study resolved this problem by directly choosing proper artificial viscosity terms.

The results for the store pressure computations carried out using the method discussed in Section V are next presented in Figures 49 to 57. It is interesting to note from these results that the panel method alone can predict the store on-body pressures reasonably well in regions near the store nose. Also, the supersonic pocket near the nose shoulder is computed well. The linear theory alone fails to compare well with the experiments due to the interference caused by the presence of the pylon and wing. Whereas the present theory does a good job of predicting the two accelerative and decelerative behaviors

in the center and aft portions of the store. Overall favorable agreement between the present computational scheme and the experimental results has been observed along the bottom periphery of the store. As the free stream Mach number has been increased from free stream subsonic to supersonic values, the acceleration flow near the nose shoulder apparently shows a strong shock which has not been predicted well in the linear computations and hence does not show up in the present computations as well. The matching of the linear velocities alone up to the store nose shoulder, i.e. up to $\bar{x}=2.0$ and the present solution of adding the interference flow to the full potential solution could be a viable approach to obtaion a better approximation for the present problem. As mentioned in Section V, the linear addition of the influences due to the non-linear effects (both due to the governing differential equation and due to the interference effects) is not fully justifiable. However, an attempt has been herein made to try such naive methods to understand the physical significance of each of the nonlinearities. The results presented in Figures 49 to 57 indicate that the above method proves that the present method is good for a quick and a fairly accurate estimation of the store pressure distribution and a more refined approach which builds up a correlation between the linear and non-linear effects independently for the two parts and then implementing a cross correlation between the two should yield more accurate results. This methodology is under present investigation.

SECTION VII

CONCLUSIONS AND RECOMMENDATIONS

Conclusions

From the above analysis, the following conclusions are appropriate:

(1) The integral equation method of transonic non-linearity correction applied over the panel method linear velocities is adequate for the flow-field predictions at points free of very pronounced transonic interference effects.

(2) The weak viscous effects on flow field velocities for thin wings as in the present configurations are small and hence can be ignored.

(3) Significant non-linear effects caused both due to complex geometry and due to transonic flow non-linearities can be locally corrected, as discussed in this report, to yield approximate results. This approach avoids extensive computing that is otherwise necessary by a more rigorous procedure.

(4) Efforts to correct the store on-body pressure distribution by a simple procedure, as discussed in Section V, did not yield very good predictions. A more elaborate empirical approach is deemed necessary.

Recommendations

(1) The incorporation of a procedure for exact wake shape determination is needed. This could be done by iterating approximate wake shapes successively using the boundary conditions of tangential velocity continuity and normal velocity discontinuity across the wake by using the same singularity strengths on the wing for each iteration.

(2) Implementation of the weak separation effects into the panel method by the appropriate modification of the present viscous-inviscid interaction code is needed.

(3) MacCormack's recent implicit-explicit finite difference scheme needs to be implemented into the VNAP code of internal flow computation to reduce total computing time.

(4) Development of a good correlation between the center section velocity and velocities at other points around the store is needed in order to correct the velocities at the latter points in the internal flow computation.

(5) Finally, the various individual codes discussed in the different sections need to be combined to make a unified computational scheme with necessary user options.

REFERENCES

(1) Jameson, A., and Caughey, D. A., "Numerical Calculaton of the Transonic Flow Past a Swept Wing", New York University, ERDA Report No. C00-3077-140, June 1977.

(2) Jameson, A., and Caughey, D. A., "A Finite Volume Method for Transonic Potential Flow Calculations", Proceedings of AIAA 3rd Computational Fluid Dynamics Conference, Albuquerque, New Mexico, June 1977, pp. 35-54.

(3) Caughey, D. A., and Jameson, A., "Numerical Calculation of Transonic Potential Flow About Wing-Body Combinations", AIAA Journal, Vol. 17, No. 2, February 1979 , pp 175-181.

(4) Verhoff, A., and O'Neil, P. J., "Extension of FLO Codes to Transonic Flow Prediction for Fighter Configurations", Transonic Perspective Symposium, NASA/Ames Research Center, February 1981.

(5) Mason, W. H., et al., "An Automated Procedure for Computing the Three-Dimensional Transonic Flow over Wing-Body Combination Including Viscous Effects", AFFDL-TR-77-122, Vol. 1, February 1977.

(6) Chaudhuri, S. N., Balasubramanyan, C., and Sundaram, P., "Calculation of the Flow-Field Velocities of a Wing-Body-Stores Combination in Transonic Flow", AIAA paper No. 0958, AIAA/ASME 3rd Joint Thermophysics, Fluids, Plasma and Heat Transfer conference, June 1982.

(7) Hess, J. L.,"Calculation of Potential Flow about Arbitary Three-Dimensional Bodies", McDonnell Douglas Report MDC-J5679, October 1972.

(8) F. L. Heltsley and V. A. Cline, "Wing/Store flow field Measurment at Transonic Speeds using a Laser Velocimeter", AEDC-TR-79-5.

(9) Woodward, F.A., "Development of the Triplet Singularity for the Analysis of Wings and Bodies in Supersonic Flow", NASA CR 3466, 1981.

(10) Van Dyke, Milton, "Perturbation Methods in Fluid Mechanics", The Parabolic Press, 1974.

(11) Hess, J. L., "A fully Automatic Combined Potential-Flow Boundary-Layer Procedure for Calculating Viscous Effects on the Lifting and Pressure Distributions of Arbitrary Three-Dimensional Configurations", Report No. MDCJ7491, June, 1977.

(12) Kjelgaard, S. O., and Thomas, J. L.,"Comparison of Three-Dimensional Panel Methods with Strip Boundary-Layer Simulations to Experiment", NASA TM 80088.

(13) Cebeci, T. and Bradshaw, P.,"Momentum Transfer in Boundary Layers", Hemisphere Publishing Corporation, 1977.

(14) Cebeci, T., and Smith, A. M. O.,"Analysis of Turbulent Boundary Layers", Academic Press, New York 1974.

(15) Wu, J. M., et. al "Fundamental Studies of Subsonic and Transonic Flow Separation", AEDC-TR-77-103, Vol.2, December, 1977.

(16) Maskew, B., and Dvorak, F. A.,"The Prediction of $C_{L_{max}}$ Using a Separated Flow Model", Journal of the American Helicopter Society, April 1978.

(17) Melnik, R. E., "Turbulent Interactions on Airfoils at Transonic Speeds - Recent Developments, Computation of Viscous/Inviscid Interaction", AGARD -CP-291, October 1980.

(18) Cline, M.C., "VNAP: A Computer Program for Computation of Two-Dimensional Time Dependent Compressible Viscous Internal Flow", Los Alamos Rept- LA 7326.

(19) Keller, J. D., and South, J. C., "RAXBOD: A Fortran Program for Inviscid Transonic Flow over Axisymmetric Bodies", NASA-TM X 72231, February 1976.

(20) South, J. C., and Jameson, A., "Relaxation Solutions for Inviscid Axisymmetric Transonic Flow over Blunt or Pointed Bodies", AIAA Computational Fluid Dynamics Conference, Palm Springs, California, July 1973.

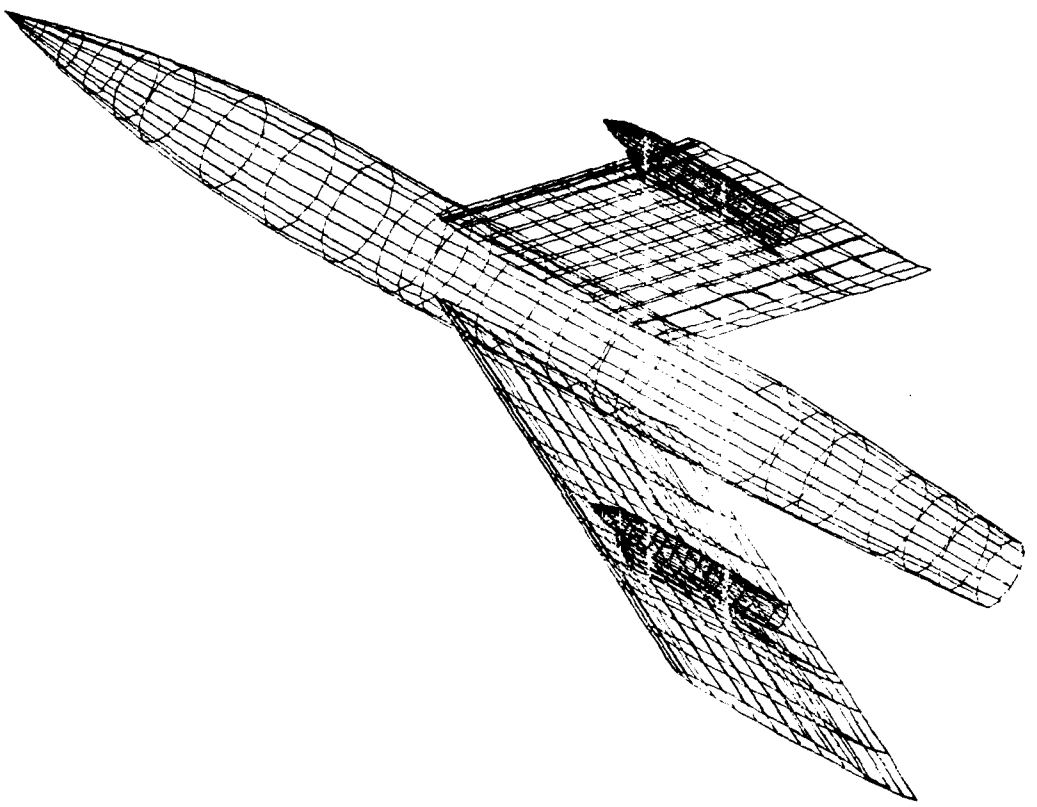


Figure 1. Typical Panelling of the Test Configuration

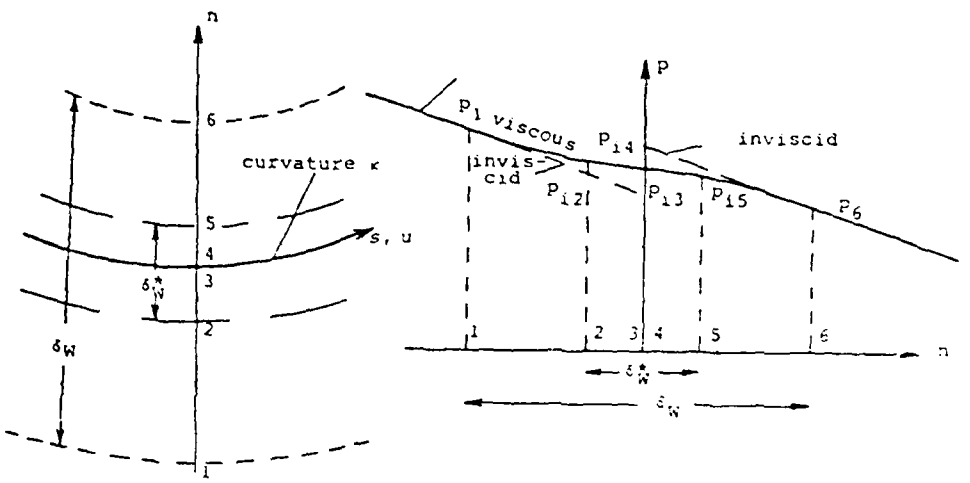
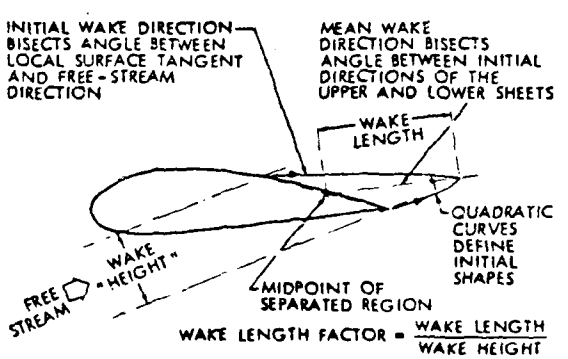


Figure 2. Wake Displacement and Wake Curvature Effects



W_H Wake Height

W_L Wale Length

Figure 3. Modified Airfoil Geometry with Separation Effects Correction

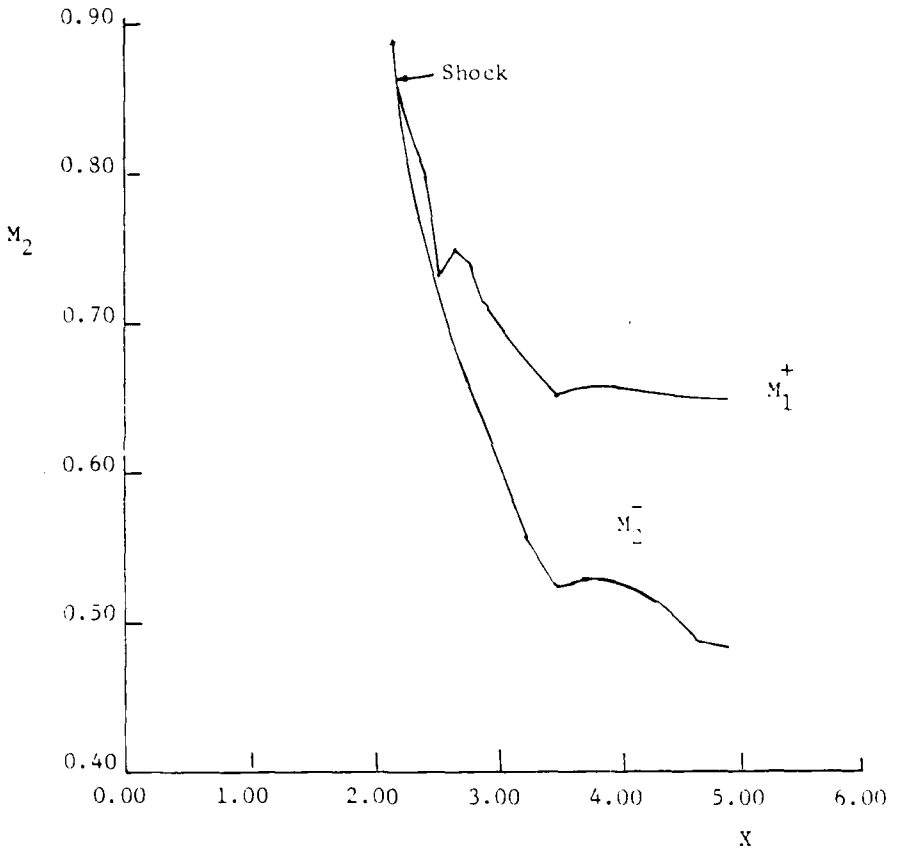


Figure 4. Shock Capturing without Displacement Effect Configuration 102, M = 0.92

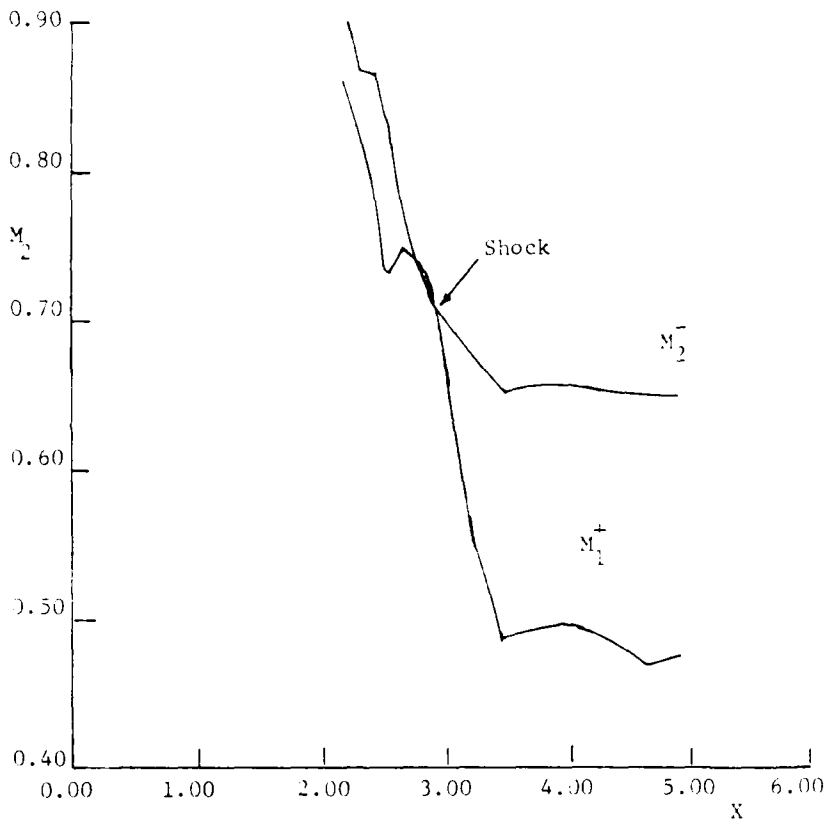


Figure 5. Shock Capturing with Displacement Thickness Added Configuration 102, M = 0.92

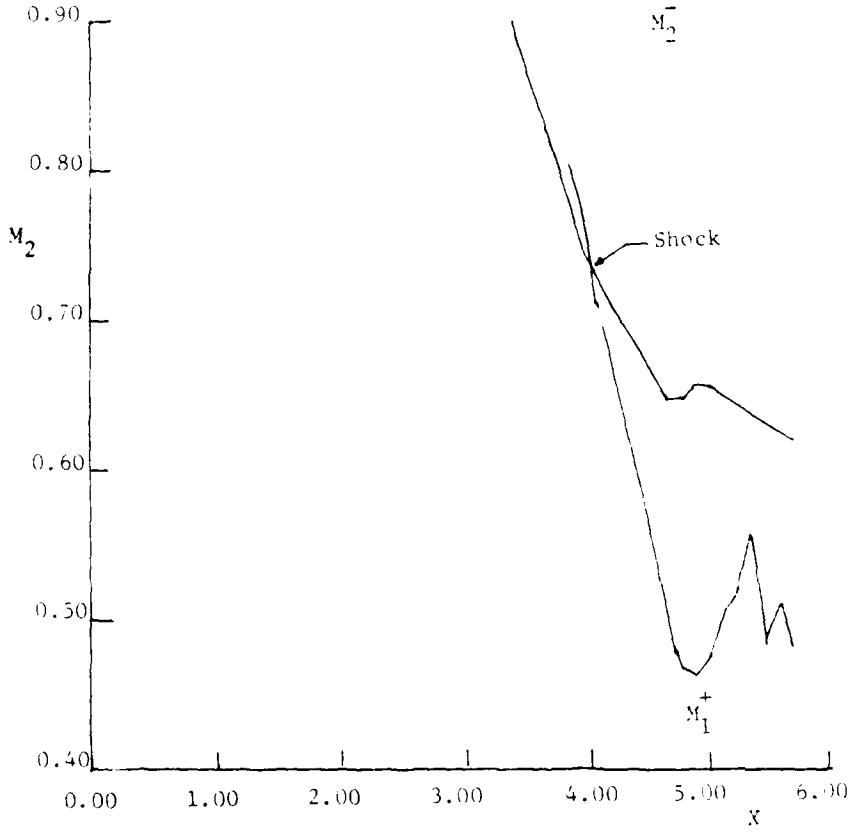


Figure 6. Shock Capturing for Configuration 101, M = 0.975

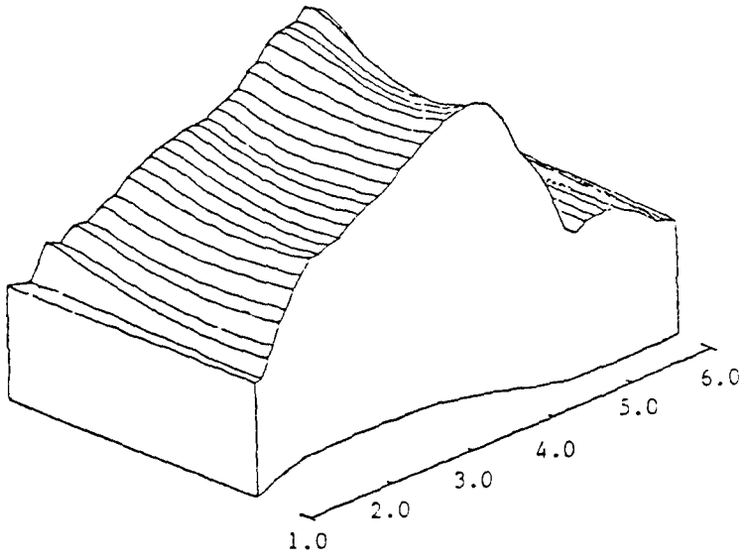


Figure 7. Velocity Surface Showing the Shock in the Gap for Configuration 101

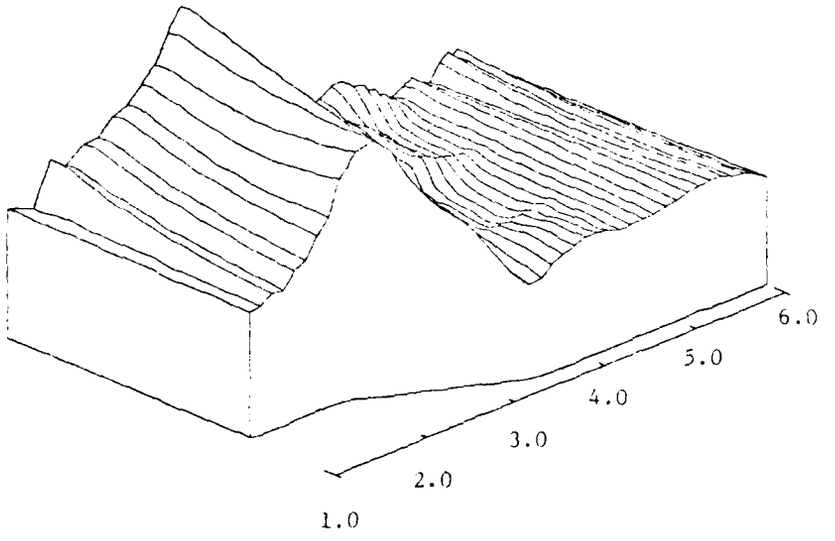


Figure 8. Velocity Surface Showing the Shock in the Gap for Configuration 102

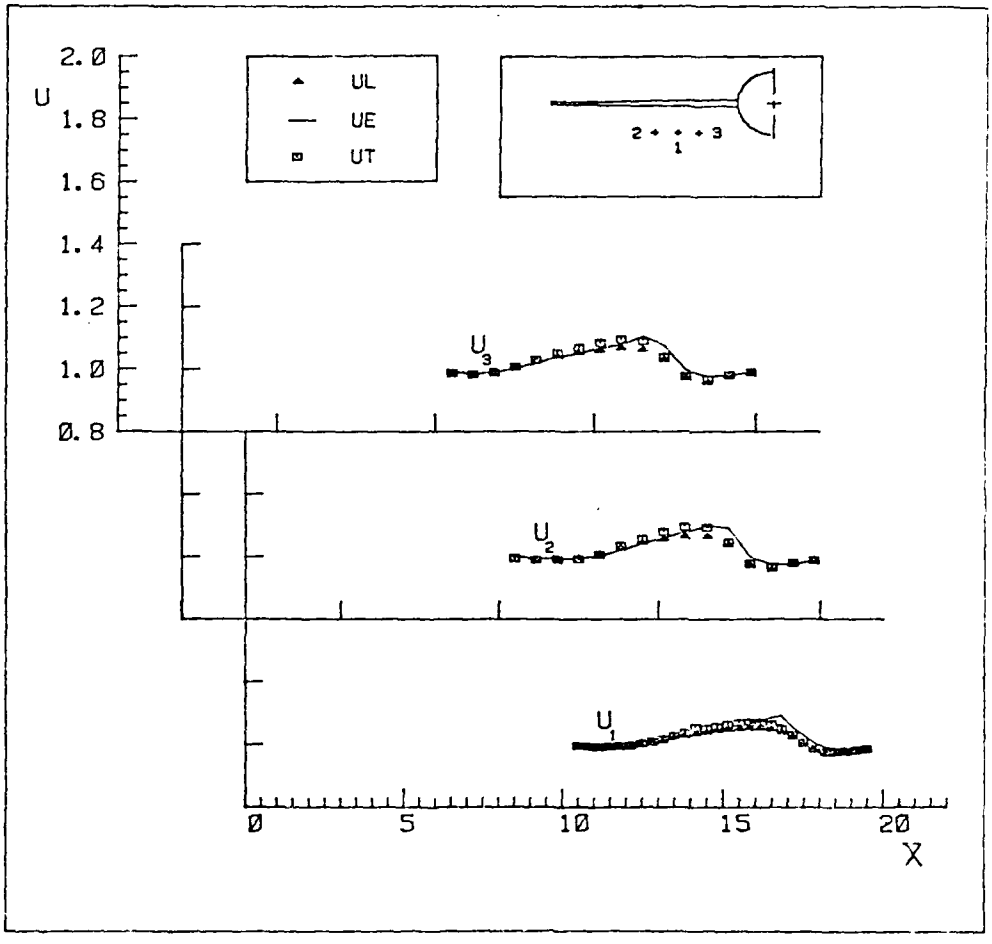


Figure 9. U VELOCITIES FOR M = Ø.925 ; ALPHA = Ø deg

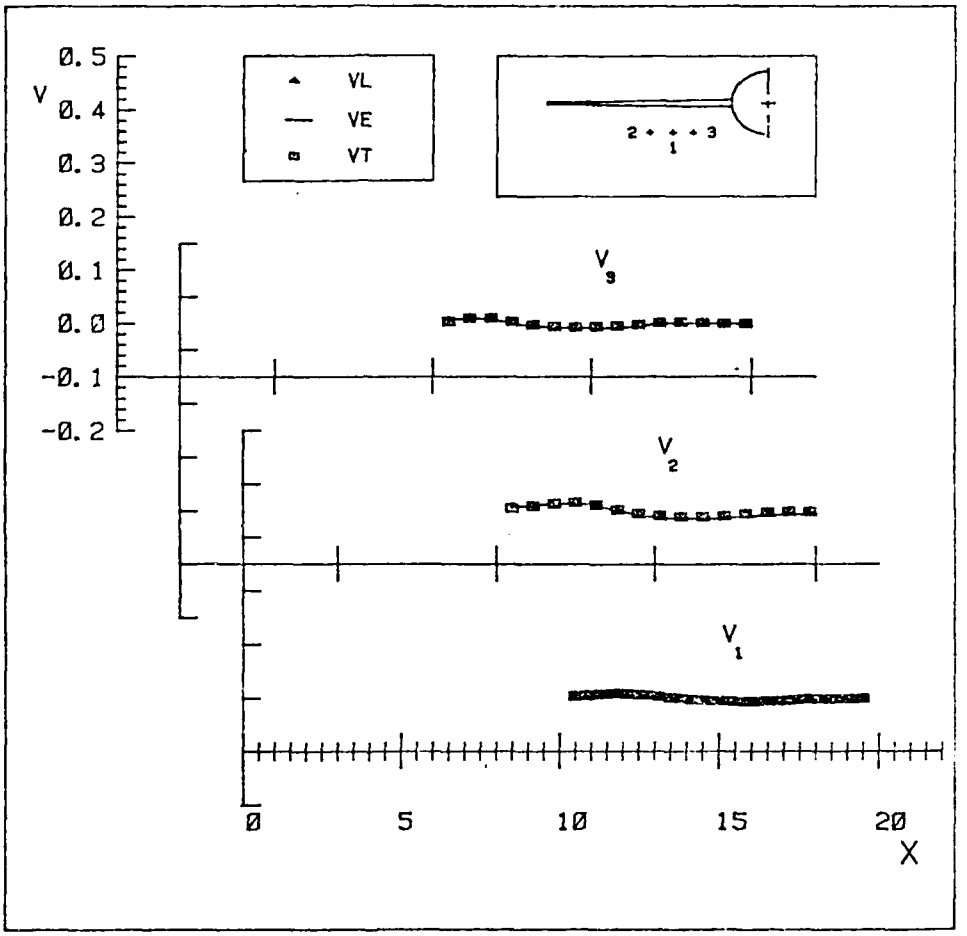


Figure 10. V VELOCITIES FOR M = 0.925 ; ALPHA = 0 deg

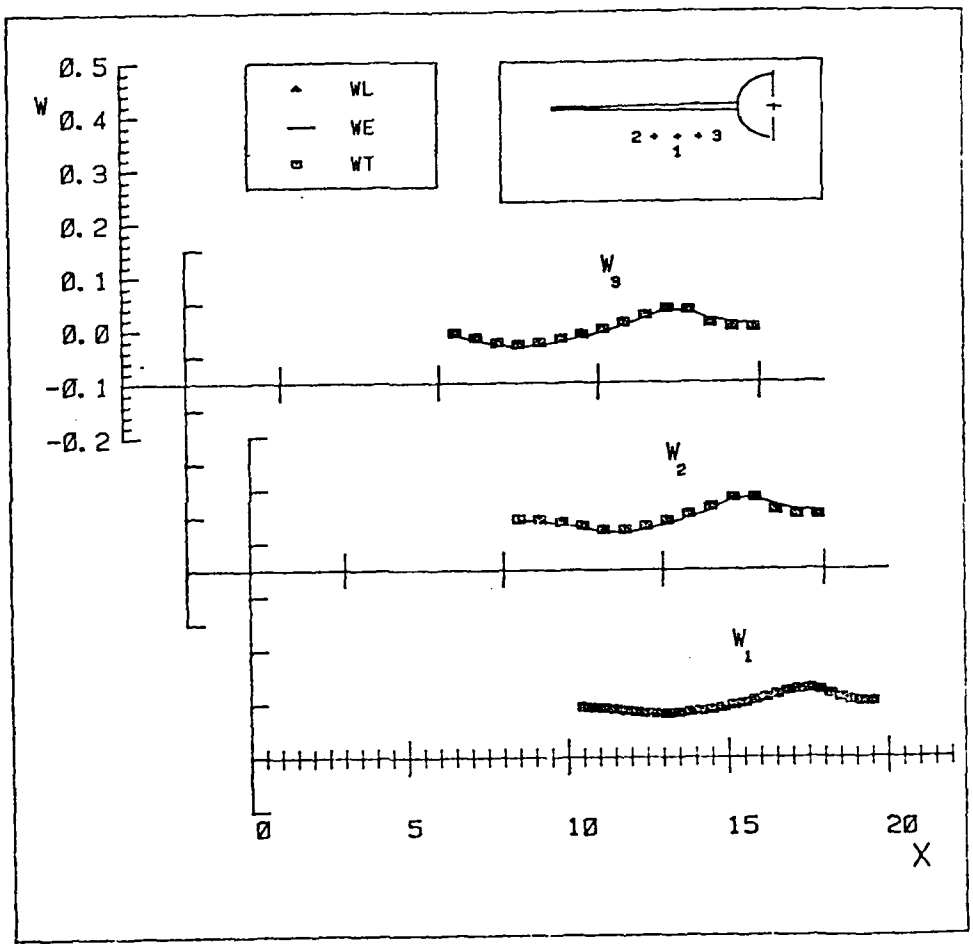


Figure 11. W VELOCITIES FOR M = Ø.925 ; ALPHA = Ø deg

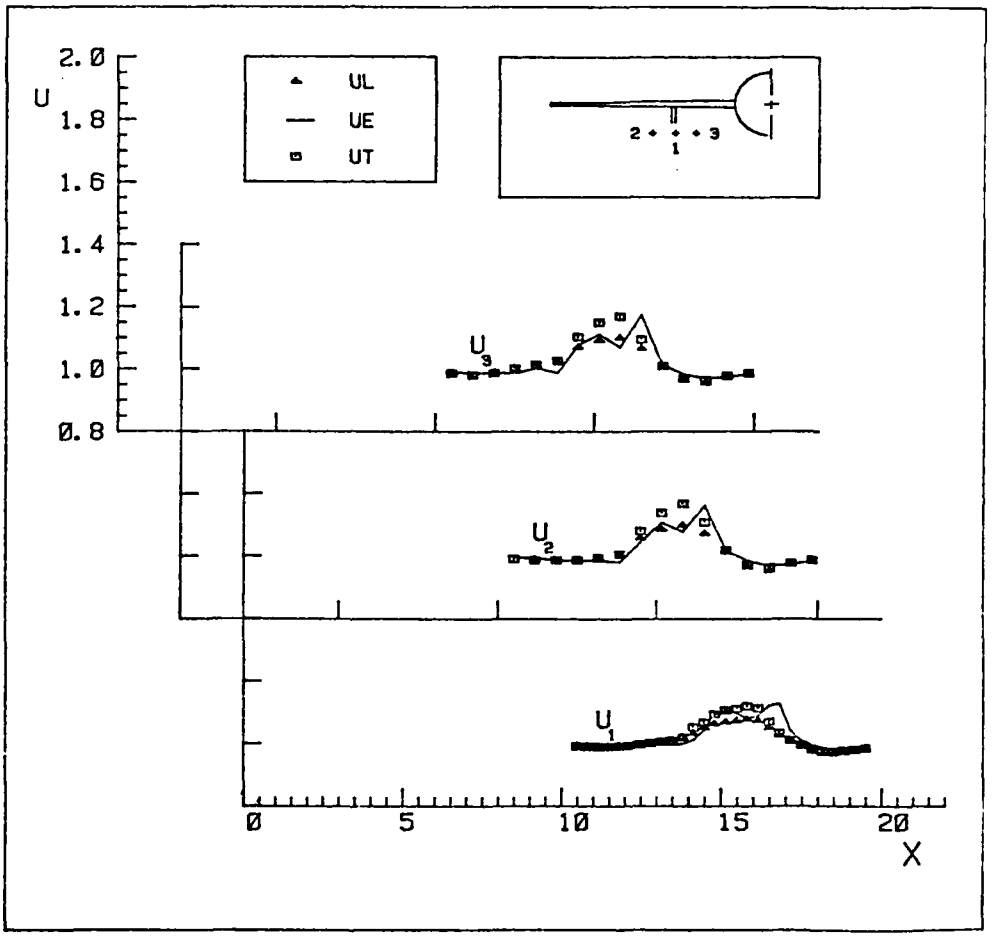


Figure 12. U VELOCITIES FOR M = 0.925 ; ALPHA = 0 deg

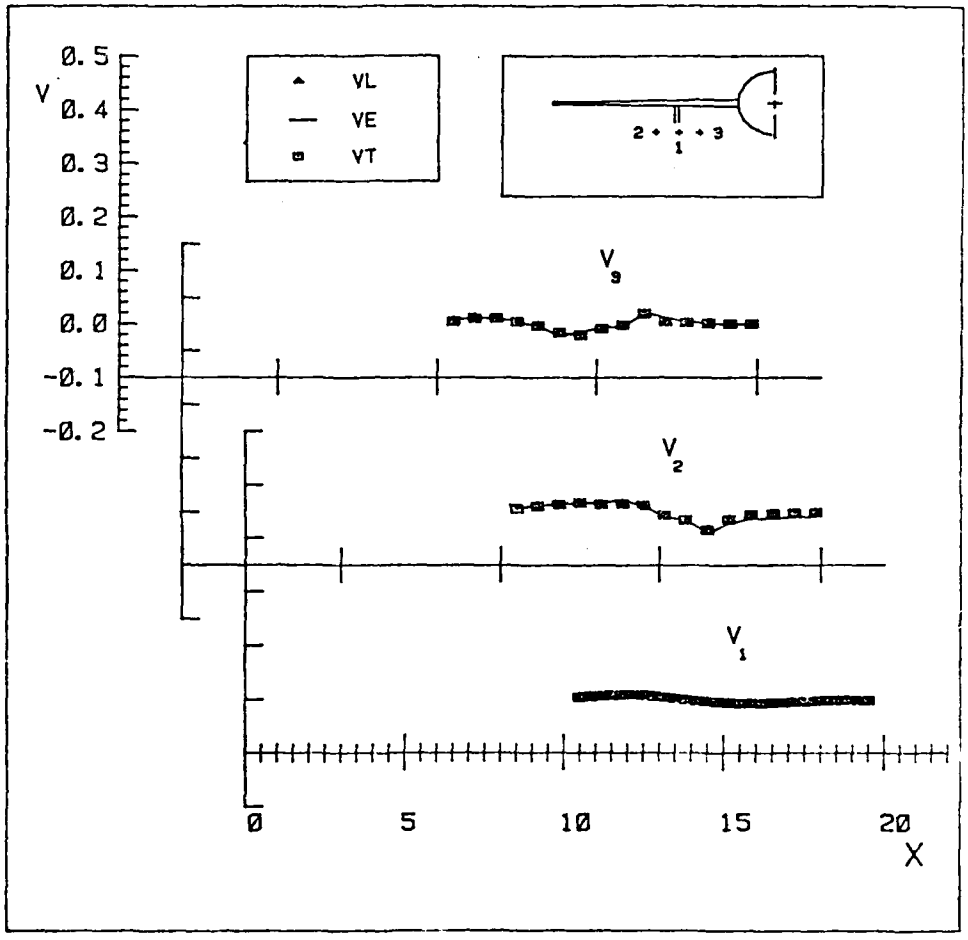


Figure 13. V VELOCITIES FOR M = 0.925 ; ALPHA = 0 deg.

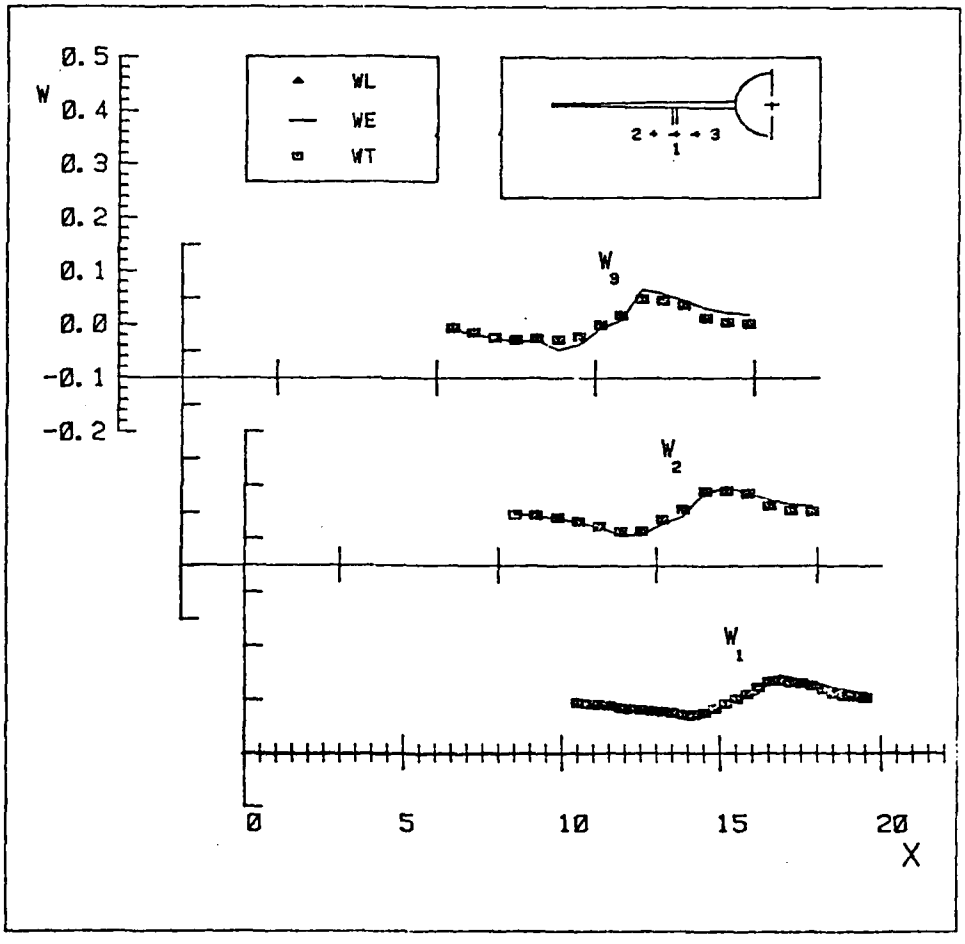


Figure 14. W VELOCITIES FOR M = 0.925 ; ALPHA = 0 deg

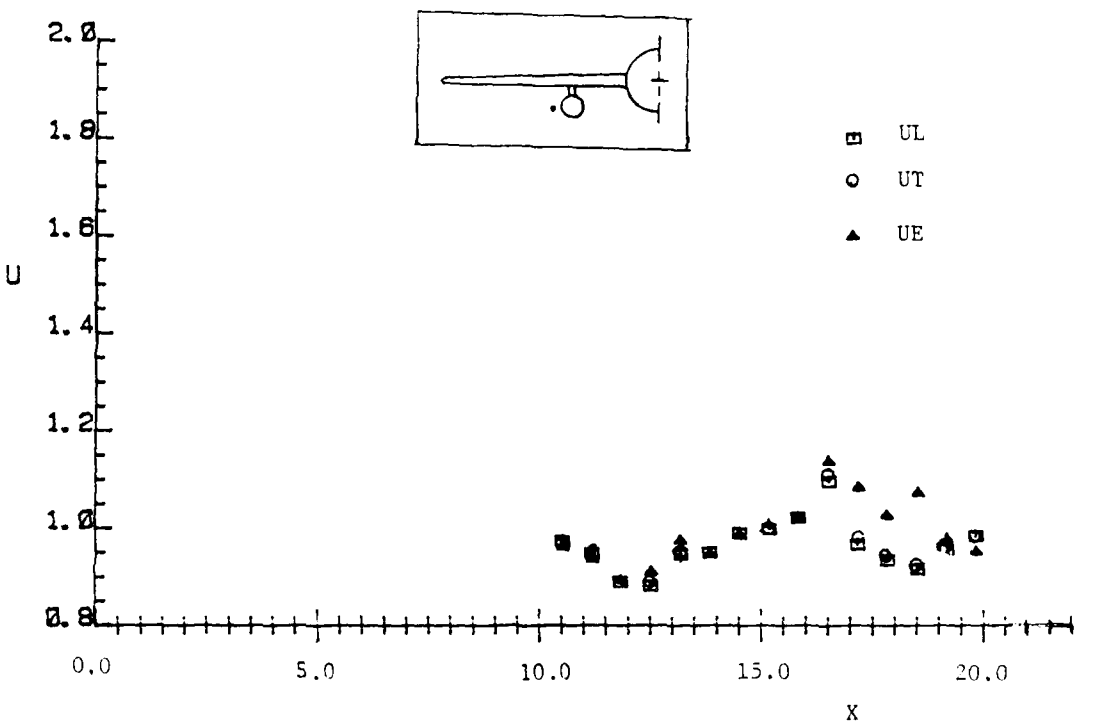


Figure 15. U Velocities at Y=4.25, Z=-1.23 for Configuration 24
M=0.925, $\alpha=5^0$

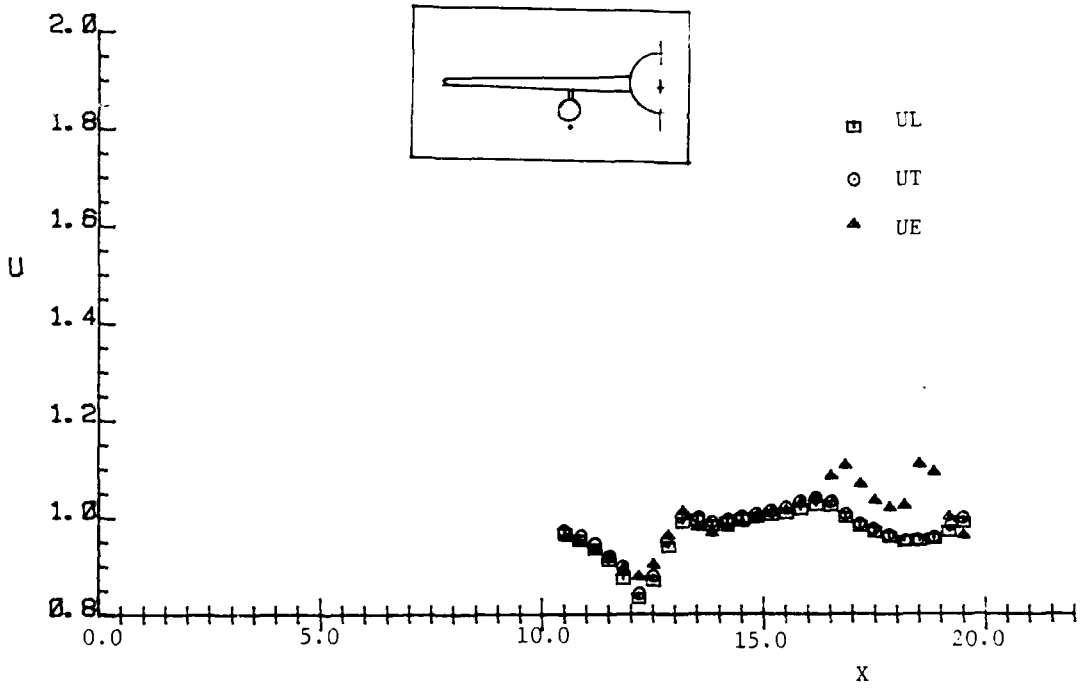


Figure 16. U Velocities at Y=3.5, Z=-1.98 for Configuration 24
M=0.925, $\alpha=5^0$

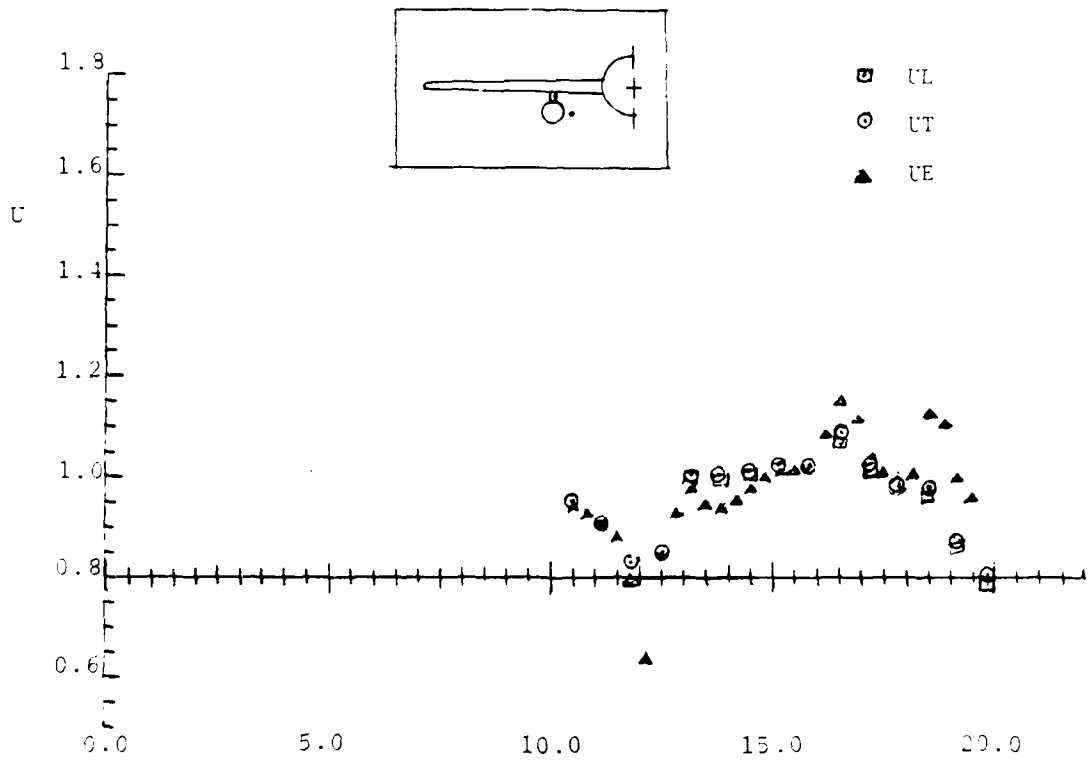


Figure 17. U Velocities at Y=2.75, Z=-1.22 for Configuration 24

M=0.925, $\alpha=5^{0}$

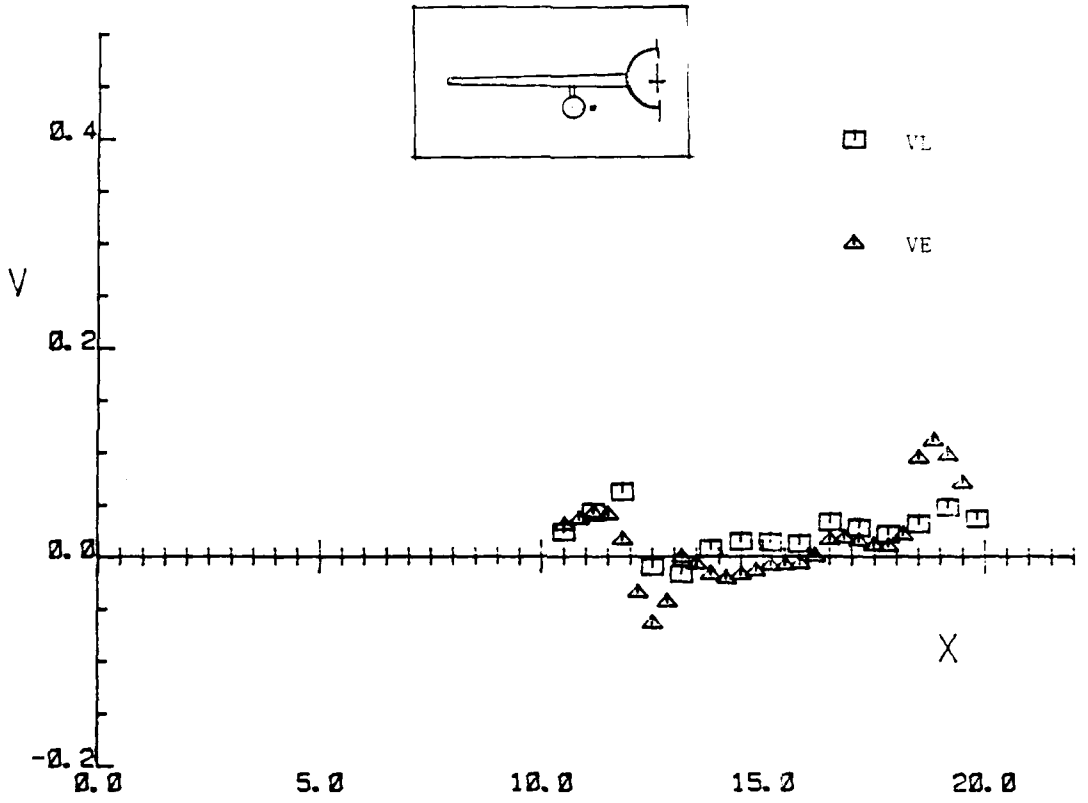


Figure 18. V Velocities at Y=2.75, Z=-1.23 for Configuration 24

M=0.925, $\alpha=5^0$

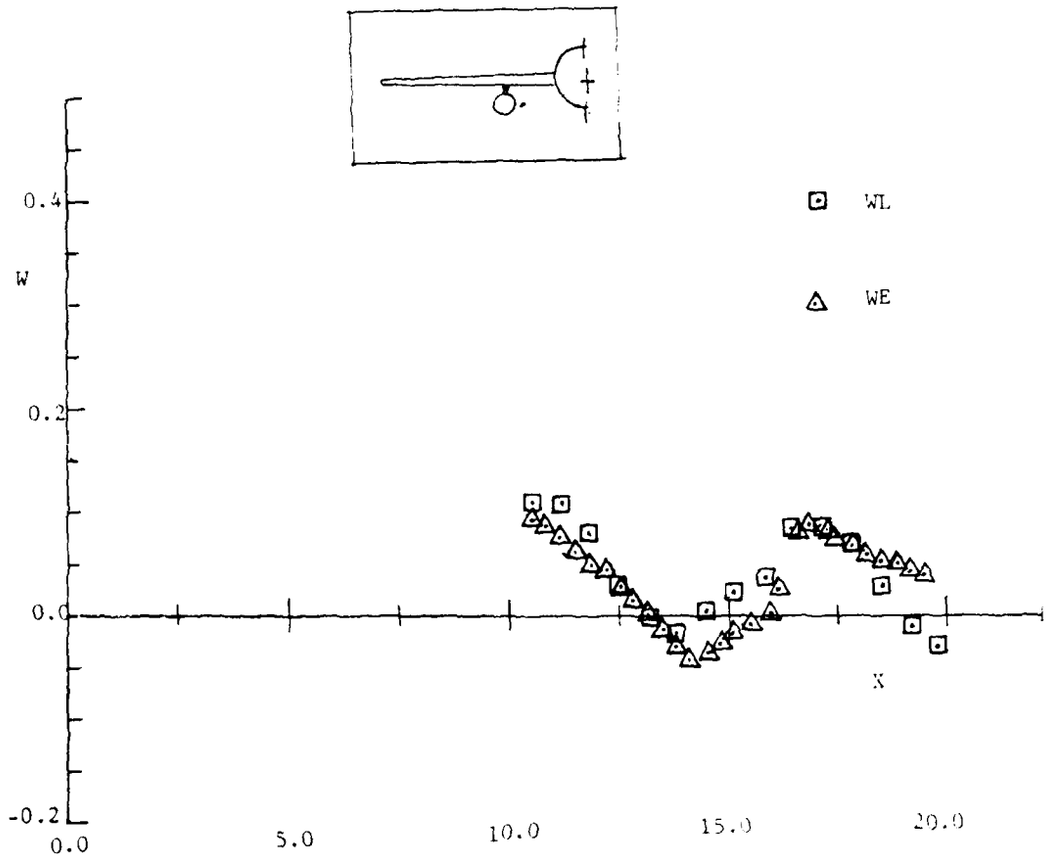


Figure 19. W Velocities at Y=2.75, Z=-1.23 for Configuration 24

M=0.925, $\alpha=5^0$

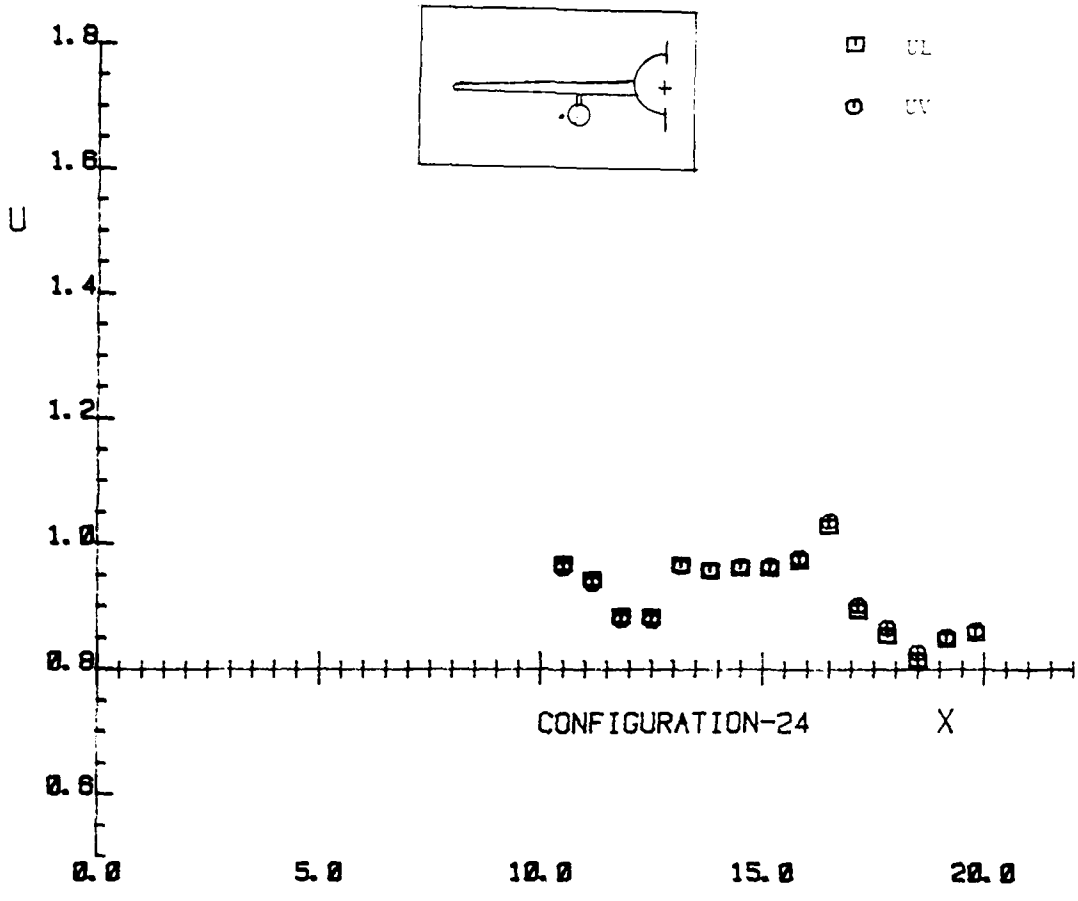


Figure 20. U Velocities at Y=4.25, Z=-1.23 for Configuration 24

M=0.925, $\alpha=5^0$

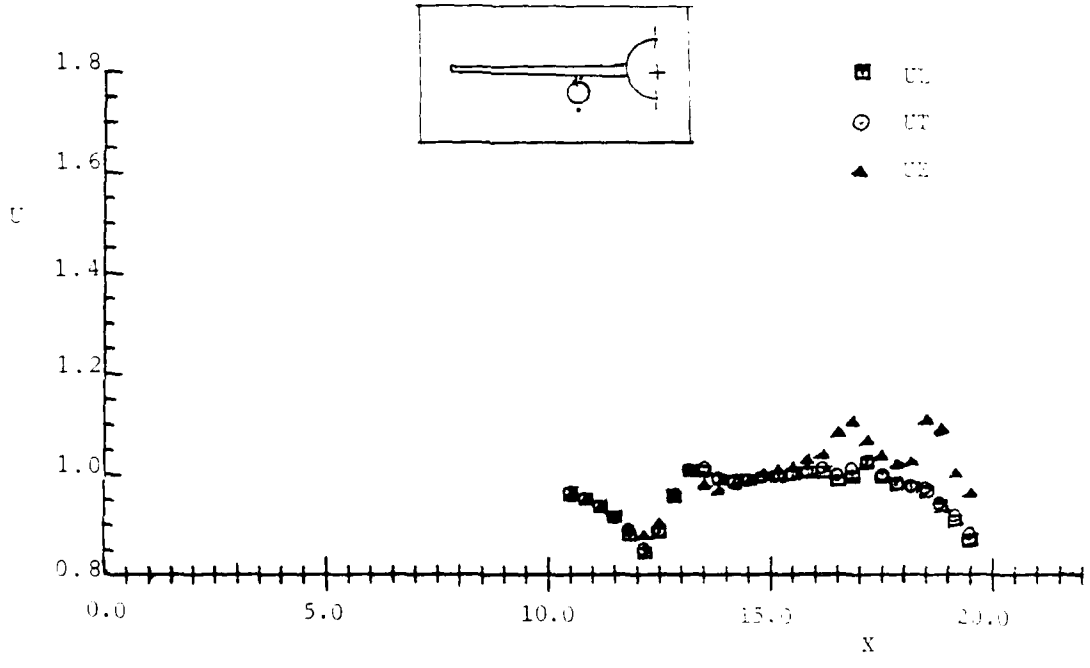


Figure 21. U Velocities at Y=3.5, Z=-1.98 for Configuration 24

M=0.925, $\alpha=5^0$

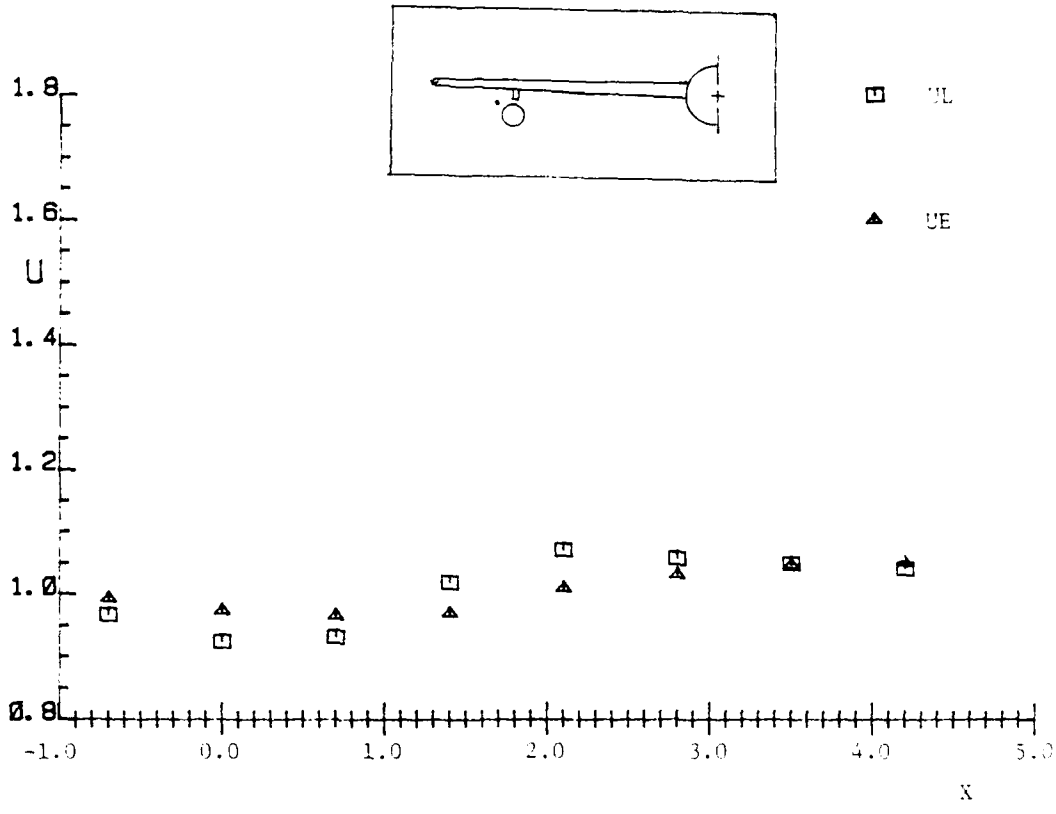


Figure 22. U Velocities at Y=-0.418, Z=0.373 for Configuration 001

M=0.70, $\alpha=2^0$

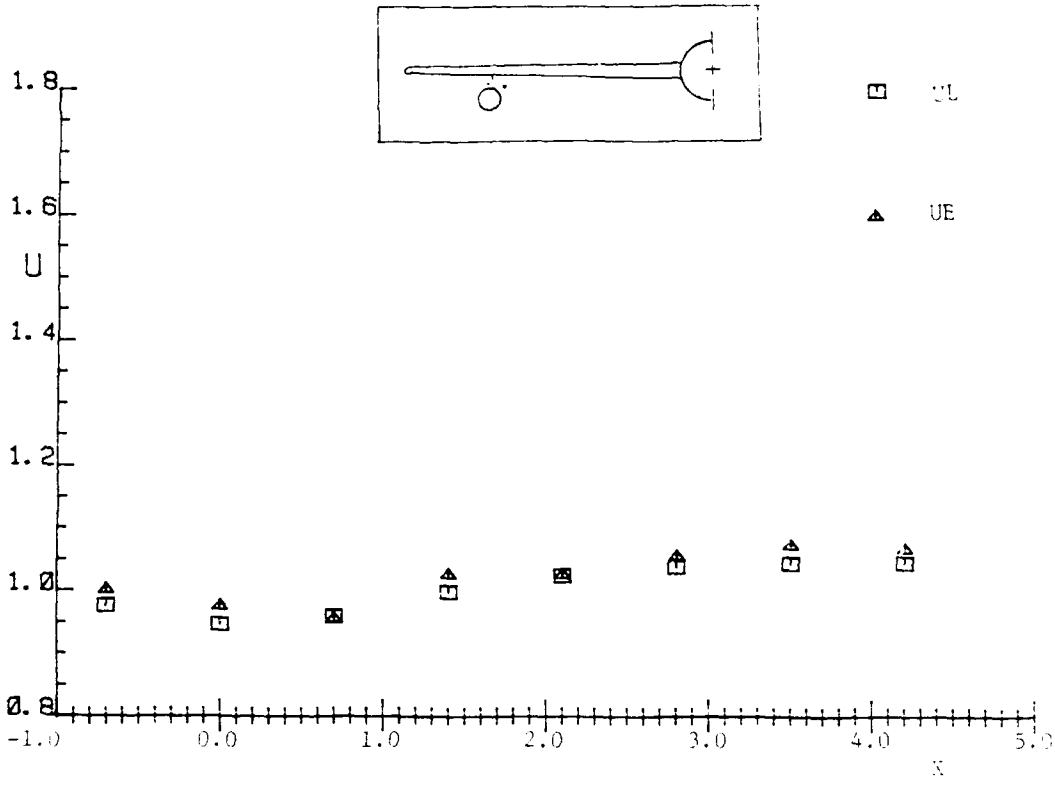


Figure 23. U Velocities at Y=0.418, Z=0.373 for Configuration 001

M=0.70, $\alpha=2^0$

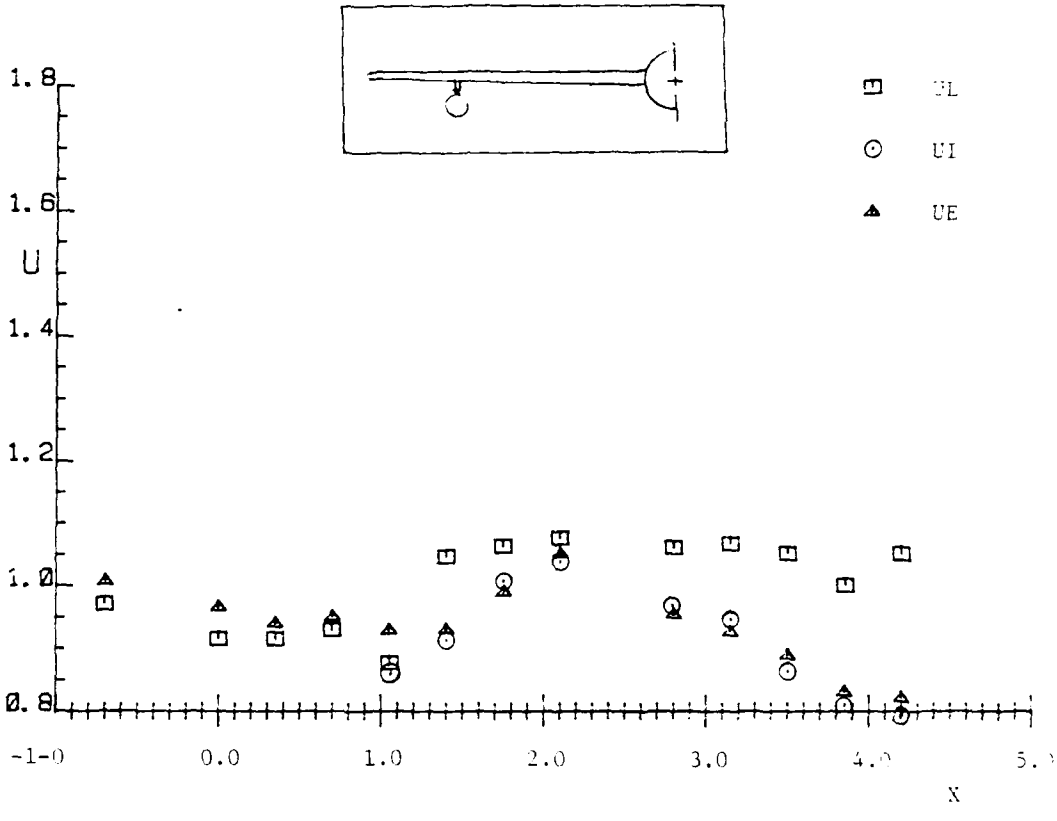


Figure 24. U Velocities at Y=0.0, Z=0.373 for Configuration 001

M=0.70, $\alpha=2^0$

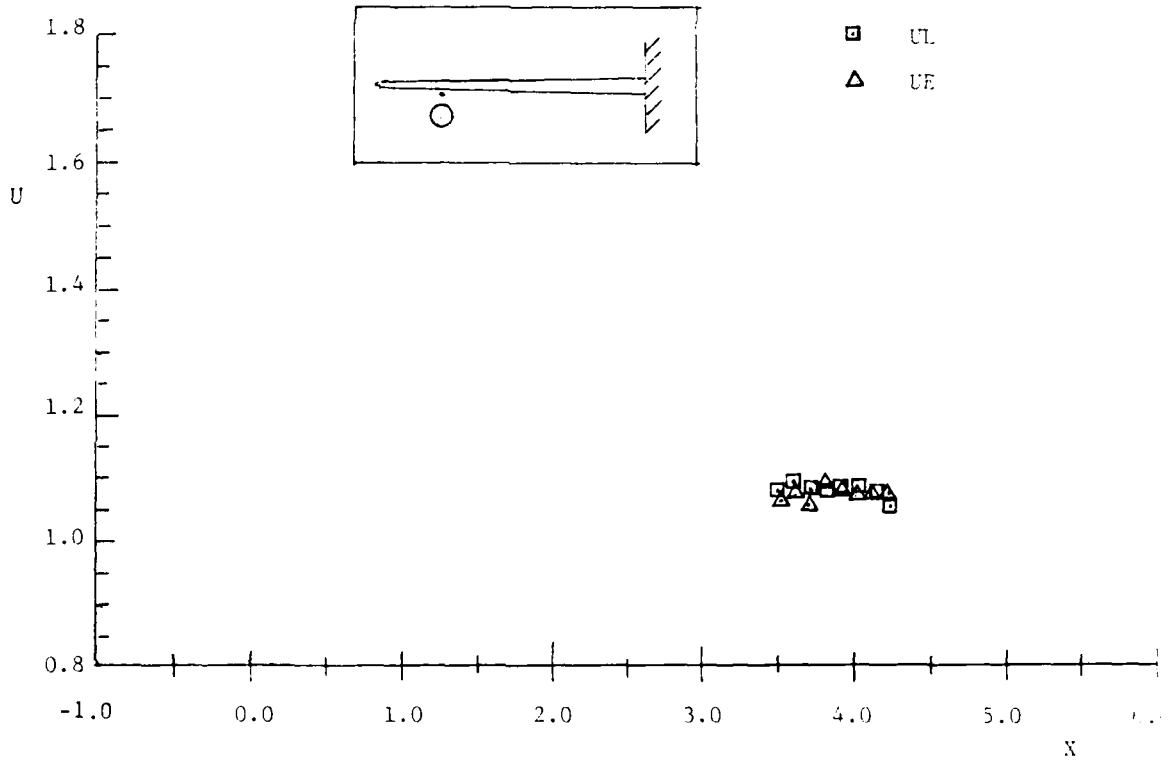


Figure 25. U Velocities at Y=0.0, Z=0.612 for Configuration 201

M=0.92, $\alpha=2^0$

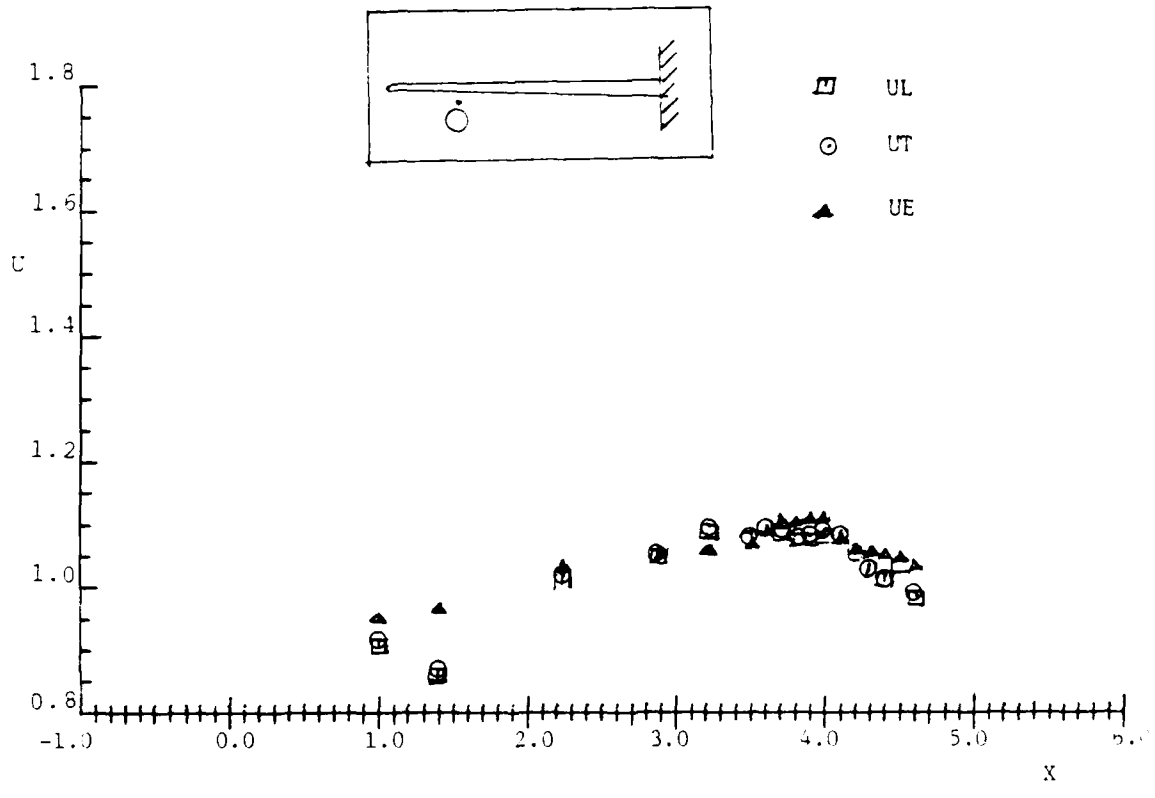


Figure 26. U Velocities at Y=0.0, Z=0.525 for Configuration 201

M=0.92, $\alpha=2^0$

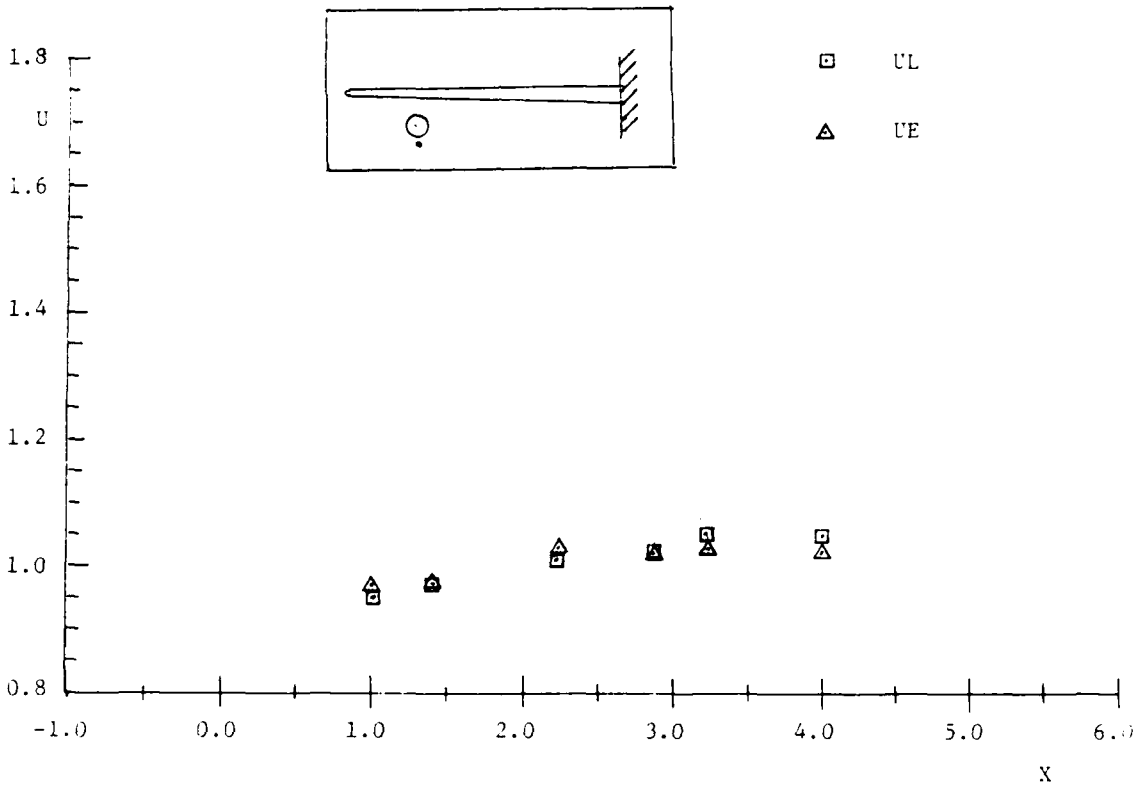


Figure 27. U Velocities at Y=0.0, Z=-0.525 for Configuration 201

M=0.92, $\alpha=2^0$

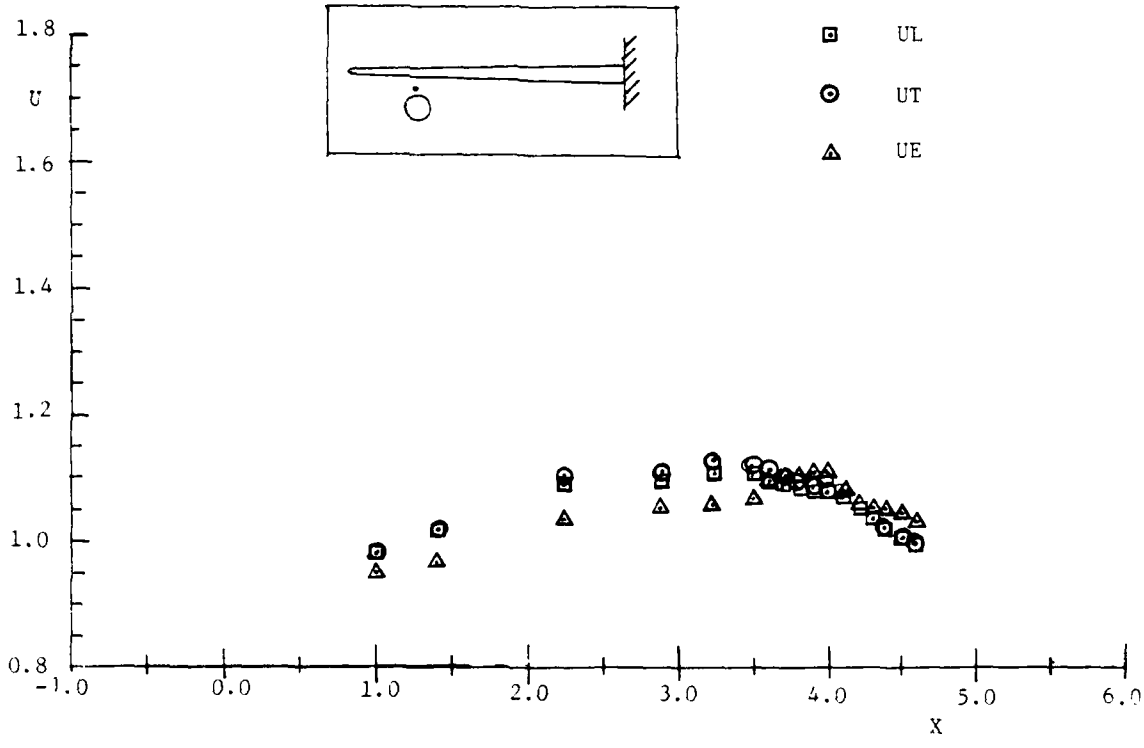


Figure 28. U Velocities at Y=0.0, Z=0.525 for Configuration 201

M=0.92, $\alpha=2^0$

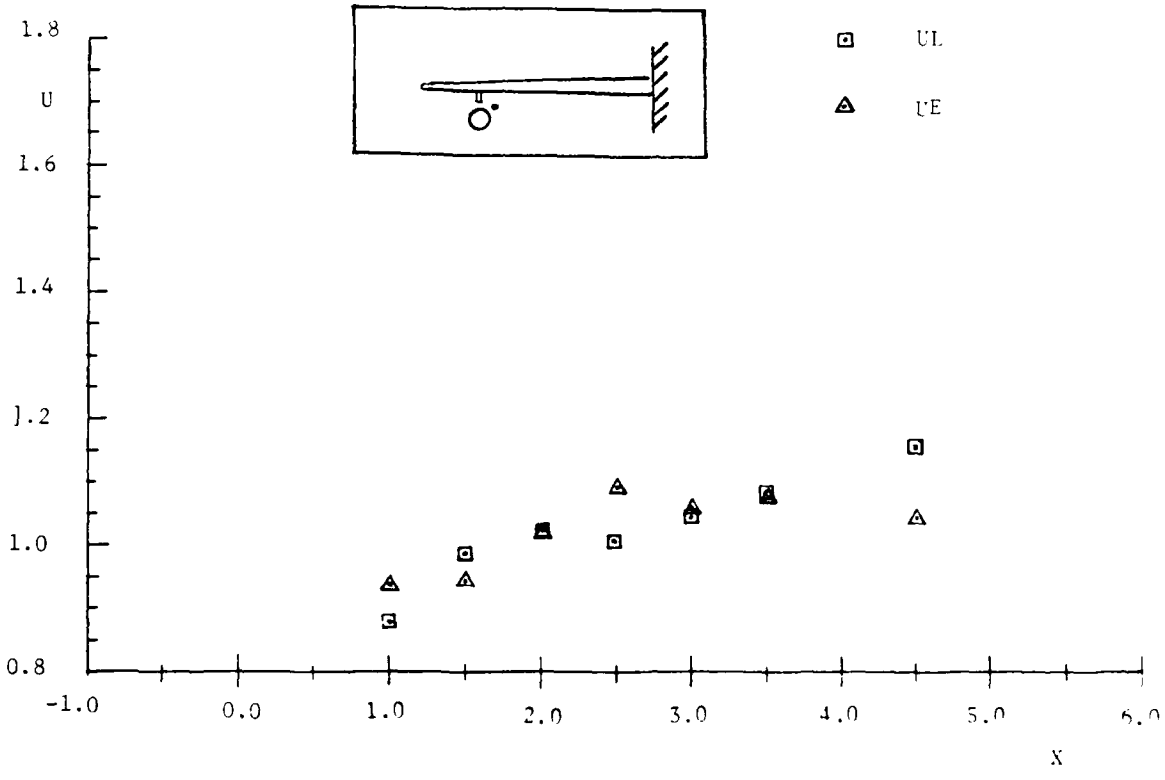


Figure 29. U Velocities at Y=0.35, Z=0.392 for Configuration 221

M=0.92, $\alpha=2^{o}$

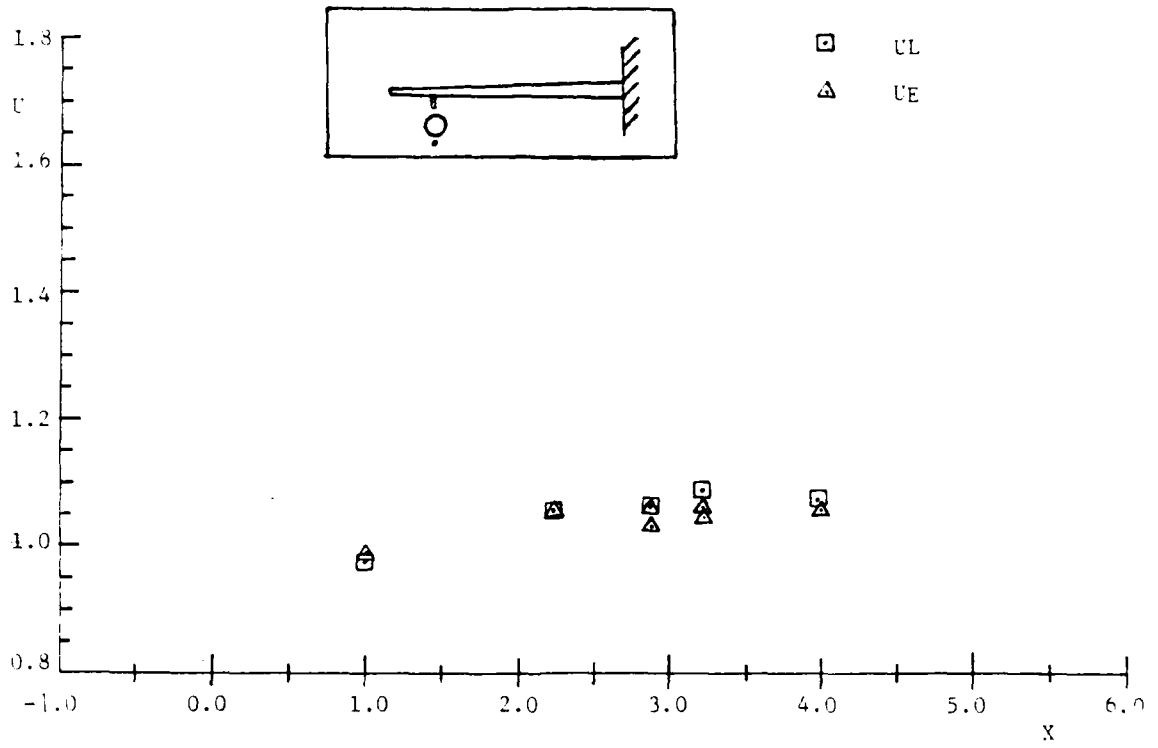


Figure 30. U **Velocities** at Y=0.0, Z=-0.525 for Configuration 221

M=0.92, $\alpha=2^0$

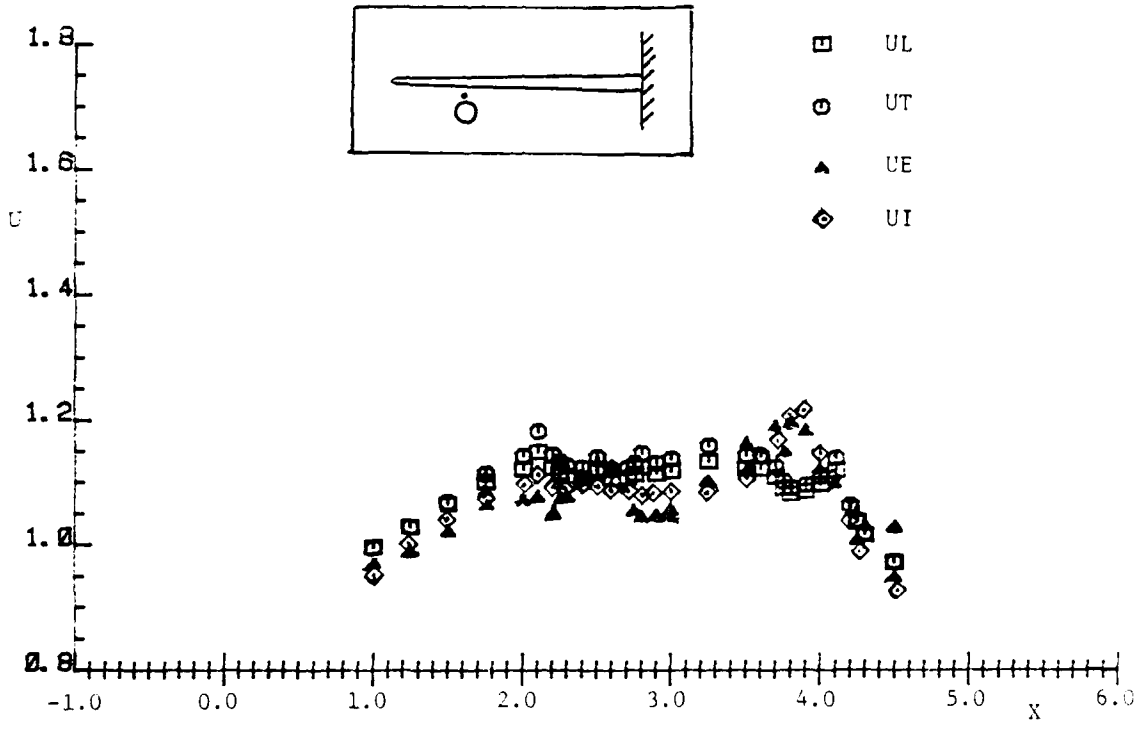


Figure 31. U Velocities at Y=0.0, Z=0.392 for Configuration 101

M=0.92, $\alpha=0^0$

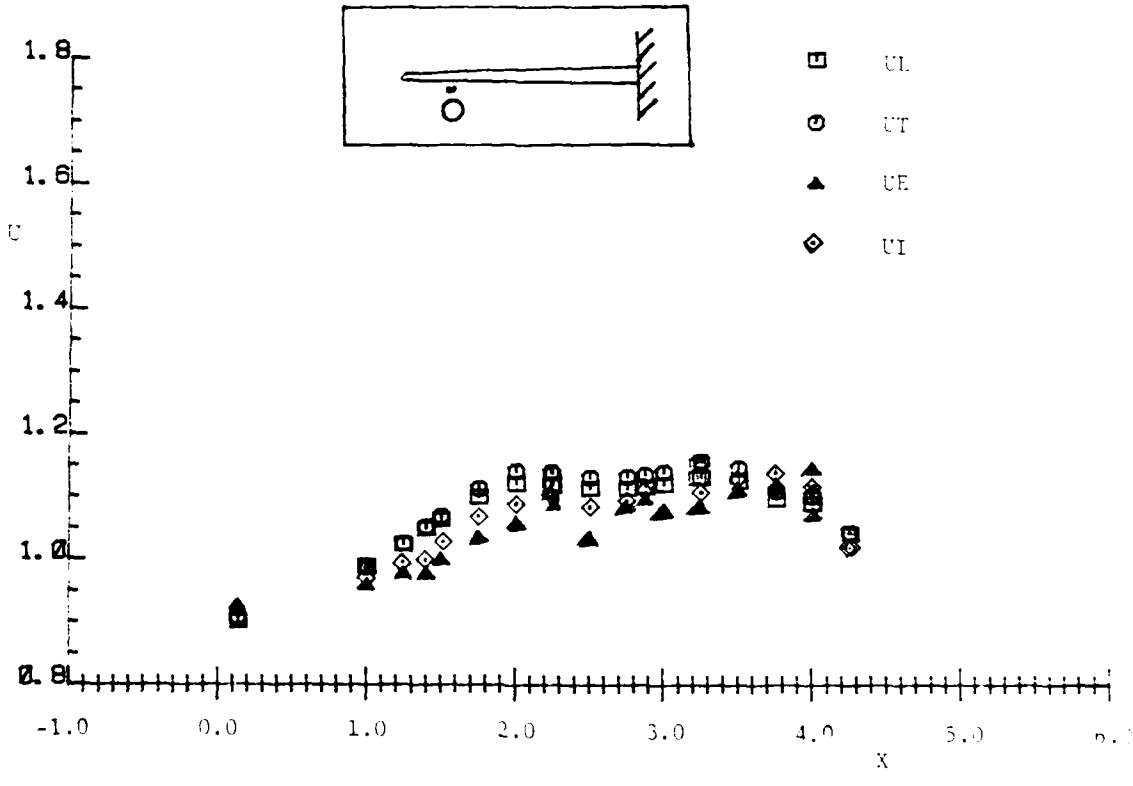


Figure 32. U Velocities at Y=0.0, Z=0.525 for Configuration 101

M=0.92, $\alpha=0^{\circ}$

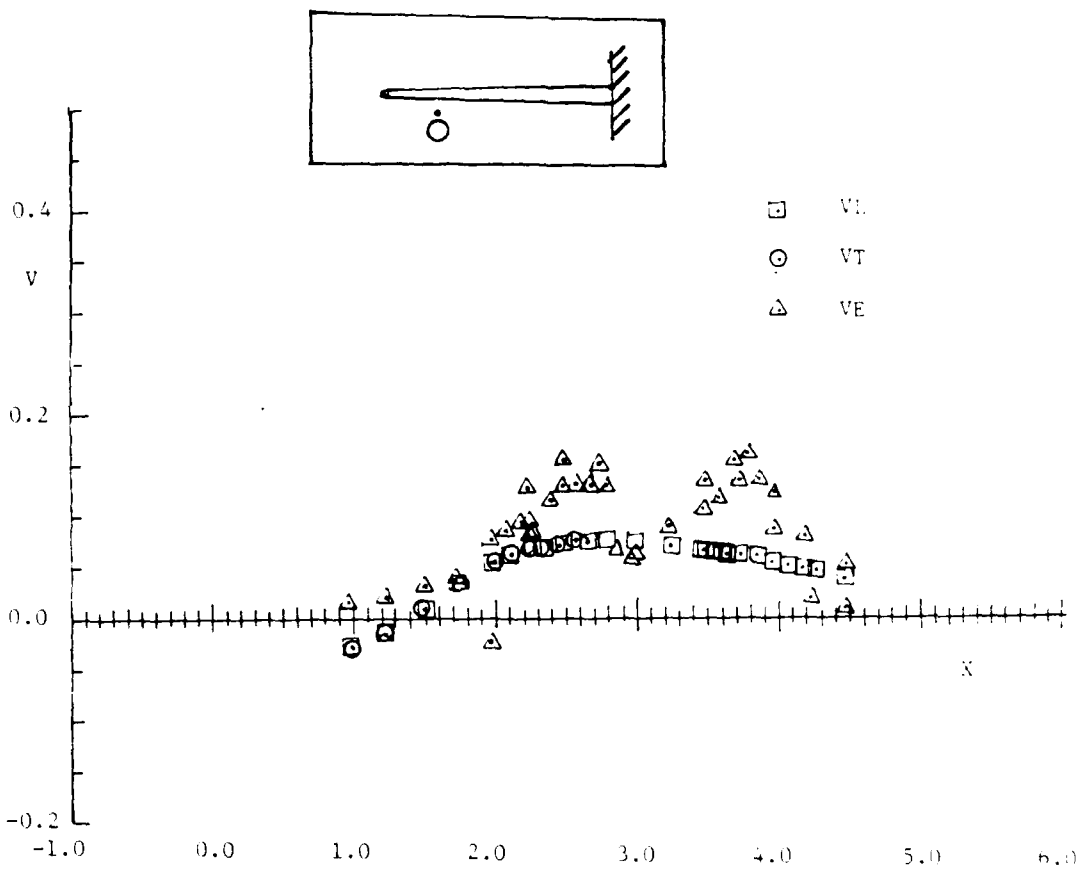


Figure 33. V Velocities at Y=0.0, Z=0.392 for Configuration 101

M=0.92, $\alpha=0^0$

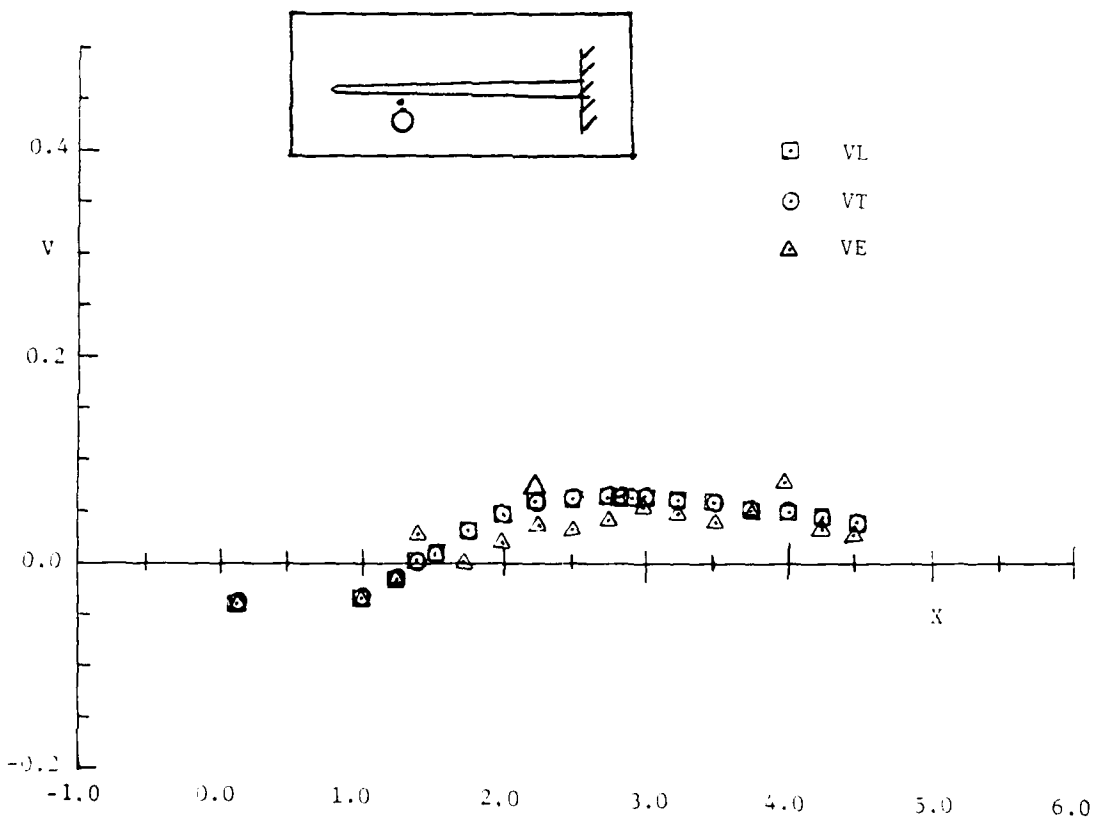


Figure 34. V Velocities at Y=0.0, Z=0.525 for Configuration 101

M=0.92, $\alpha=0^0$

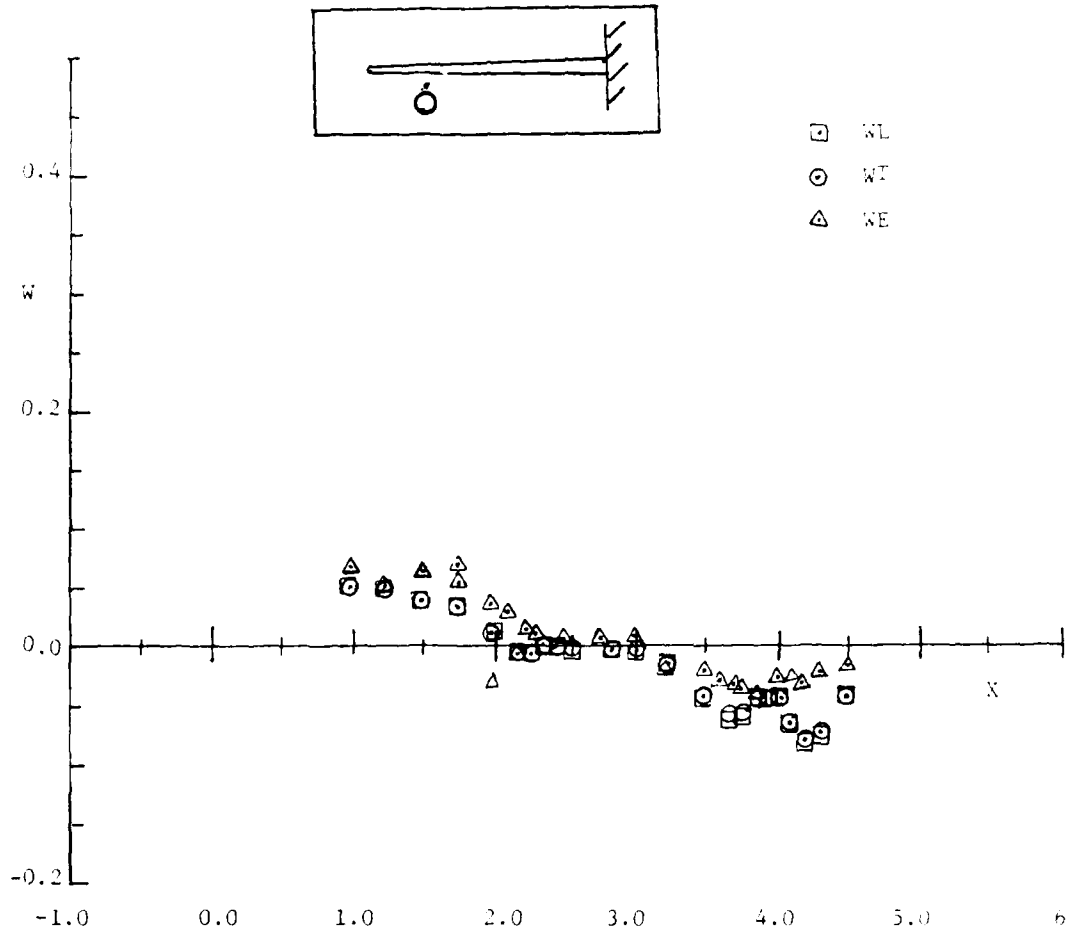


Figure 35. W Velocities at Y=0.0, Z=0.392 for Configuration 101

M=0.92, $\alpha=0^0$

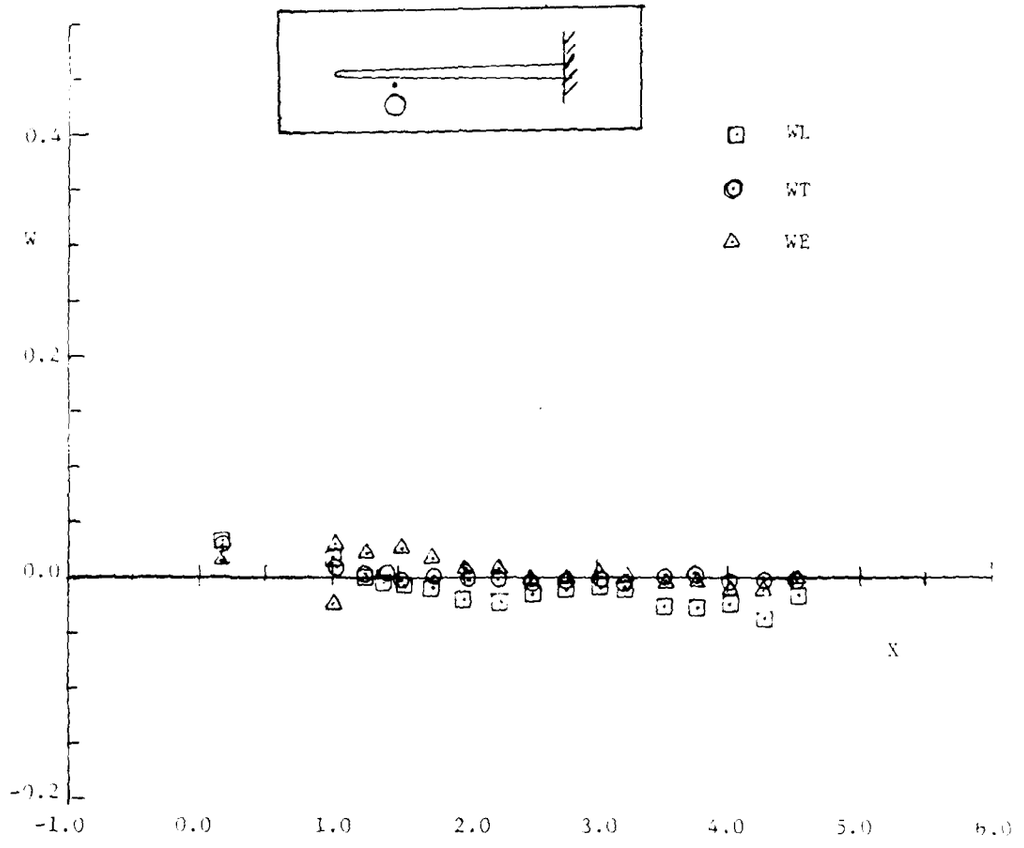


Figure 36. W Velocities at Y=0.0, Z=0.525 for Configuration 101

M=0.92, $\alpha=0^0$

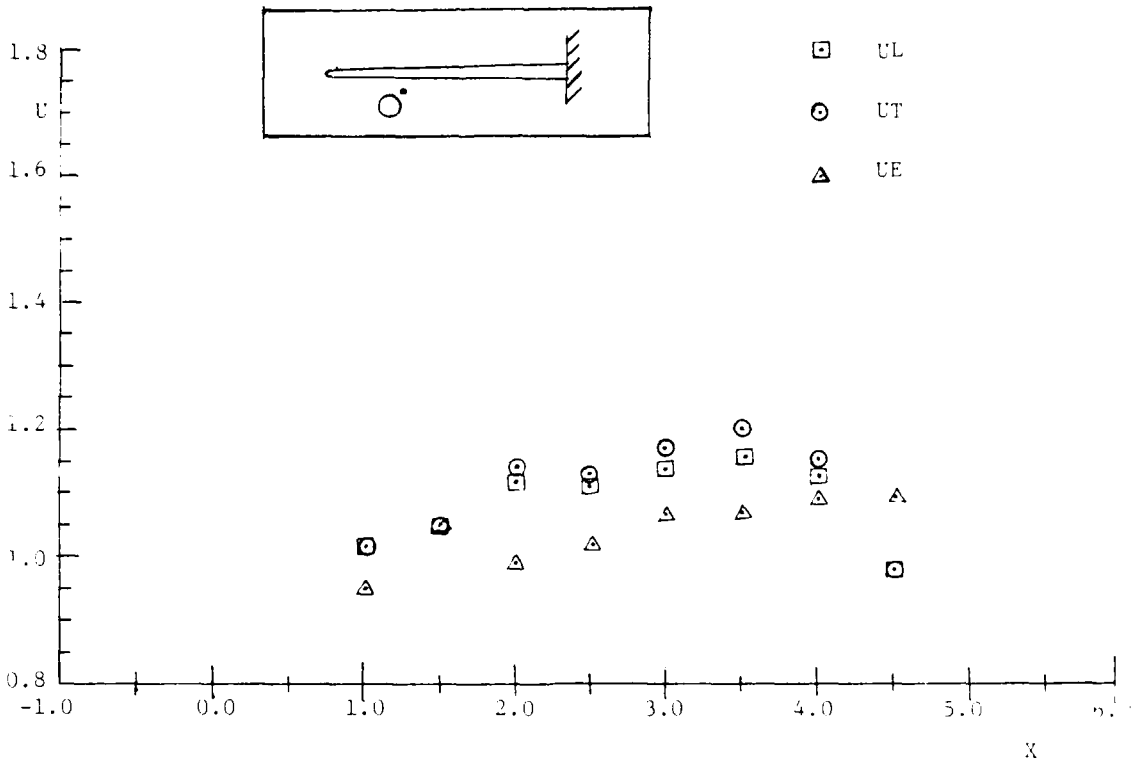


Figure 37. U Velocities at Y=0.35, Z=0.392 for Configuration 101

M=0.975, $\alpha=0^0$

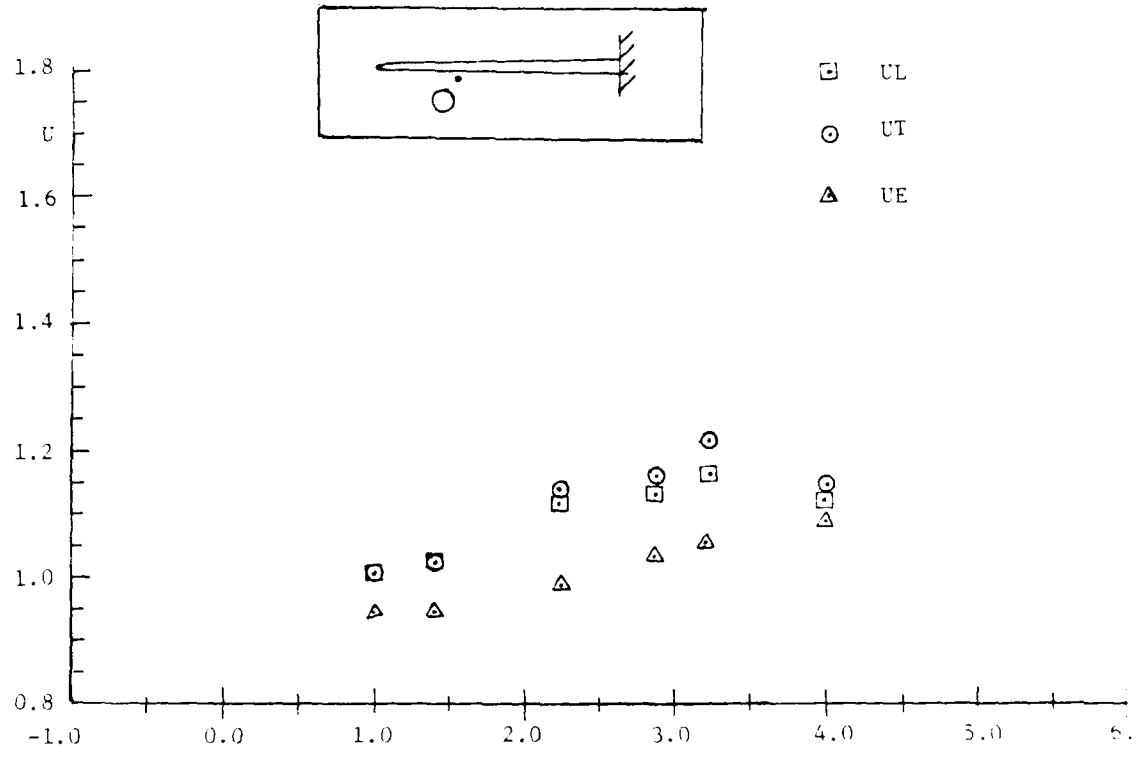


Figure 38. U Velocities at Y=0.35, Z=0.525 for Configuration 101

M=0.975, $\alpha=0^0$

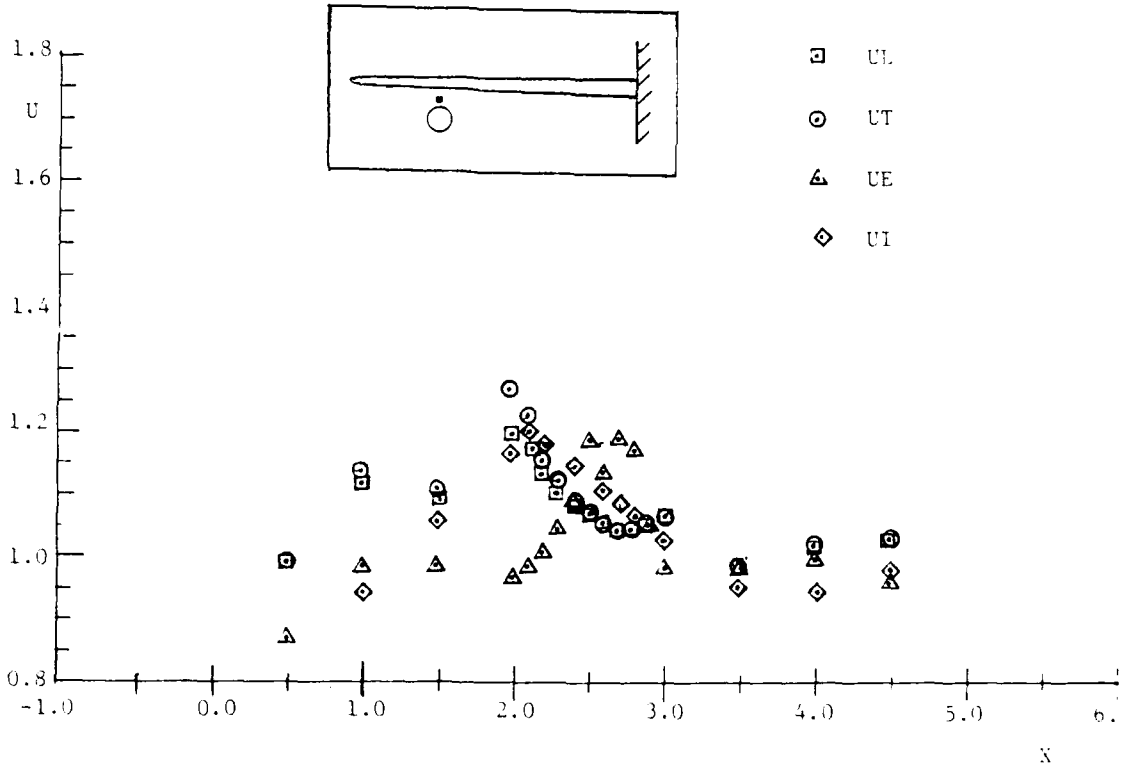


Figure 39. U Velocities at Y=0.0, Z=0.442 for Configuration 102

M=0.92, $\alpha=0^0$

(No Displacement Effect Added)

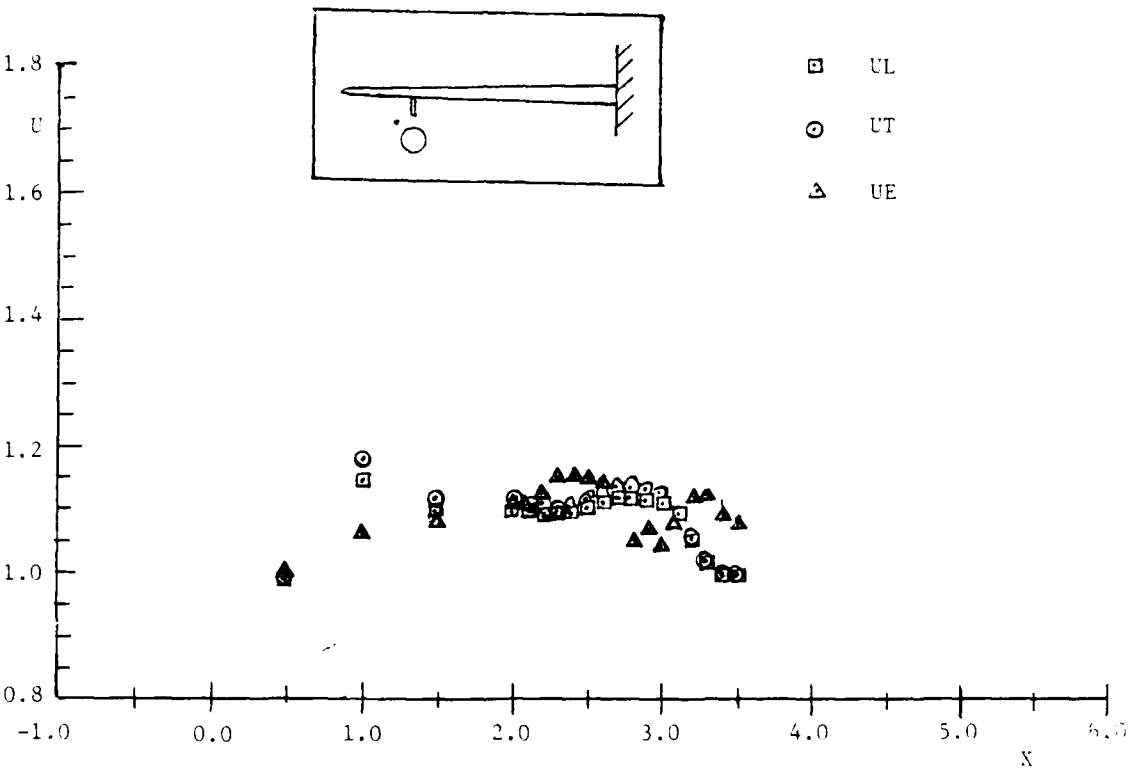


Figure 40. U Velocities at Y=-0.4, Z=0.442 for Configuration 122

M=0.92, $\alpha=0^0$

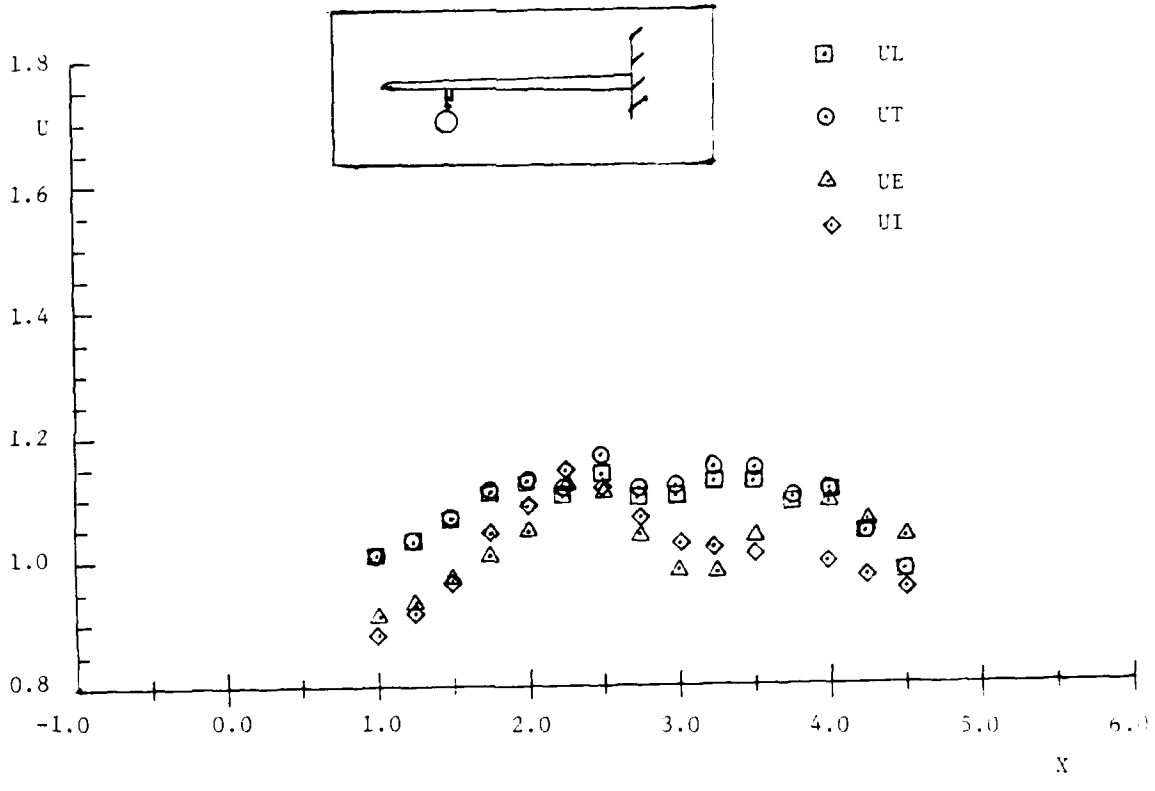


Figure 41. U Velocities in the Gap at Y=0.0, Z=0.392 for Configuration 121

M=0.92, $\alpha=0^0$

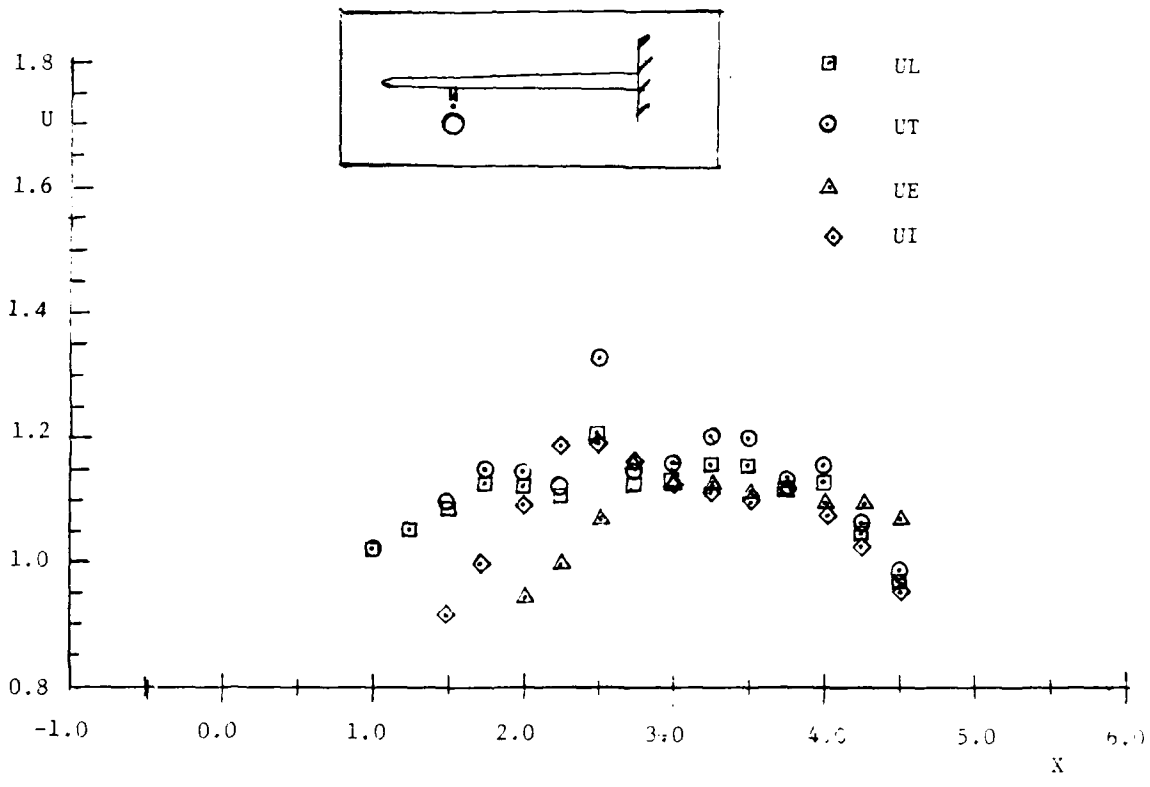
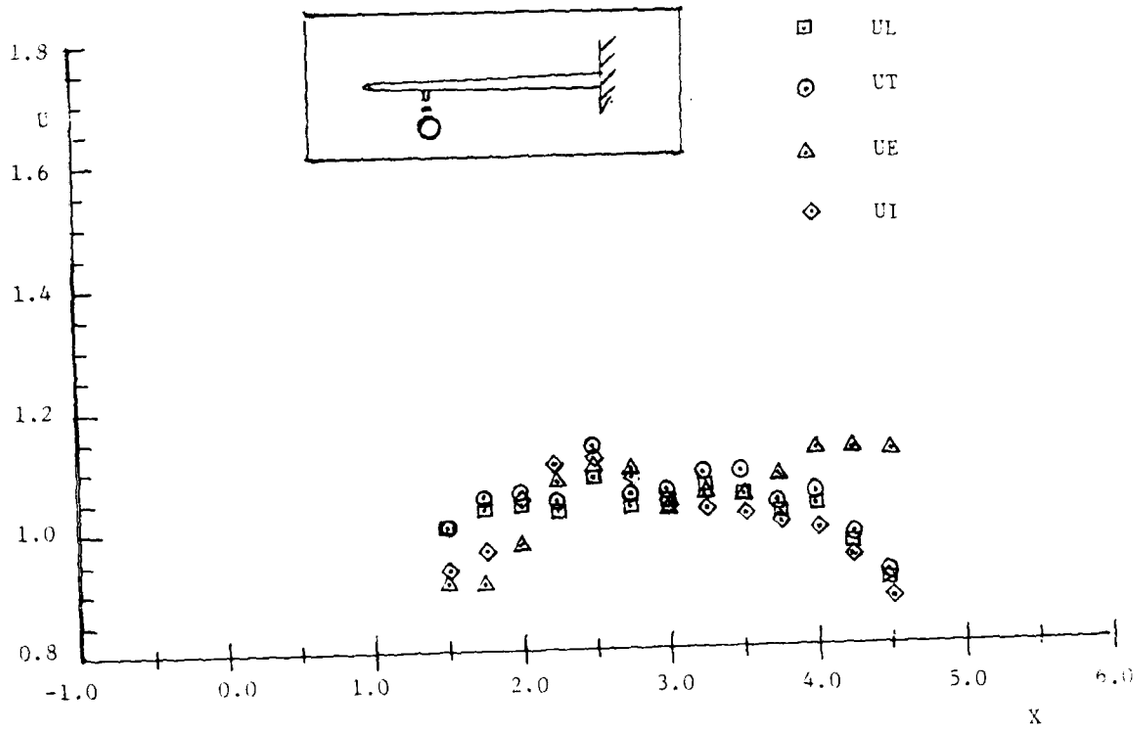


Figure 42. U Velocities in the Gap at Y=0.0, Z=0.392 for Configuration 121

M=0.975, $\alpha=0^0$



Fifure 43. U Velocities at Y=0.0, Z=0.392 for Configuration 121

M=1.025, $\alpha=0^0$

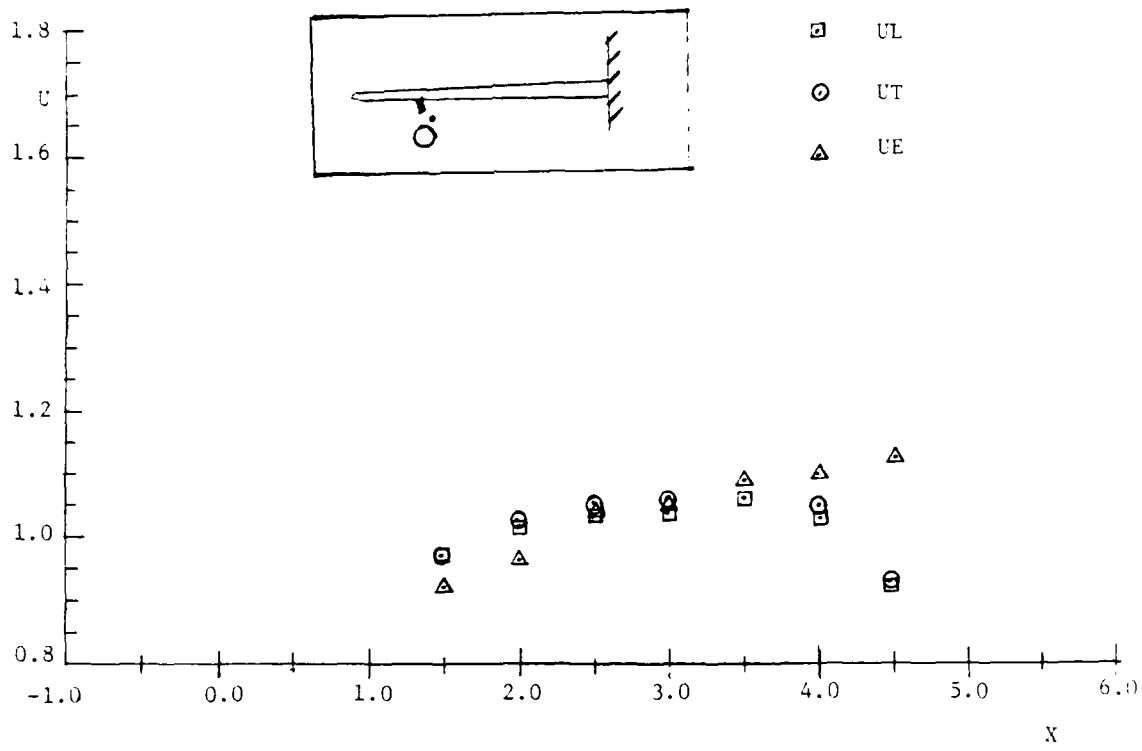


Figure 44. U Velocities at Y=0.350, Z=0.392 for Configuration 121

M=1.025, $\alpha=0^0$

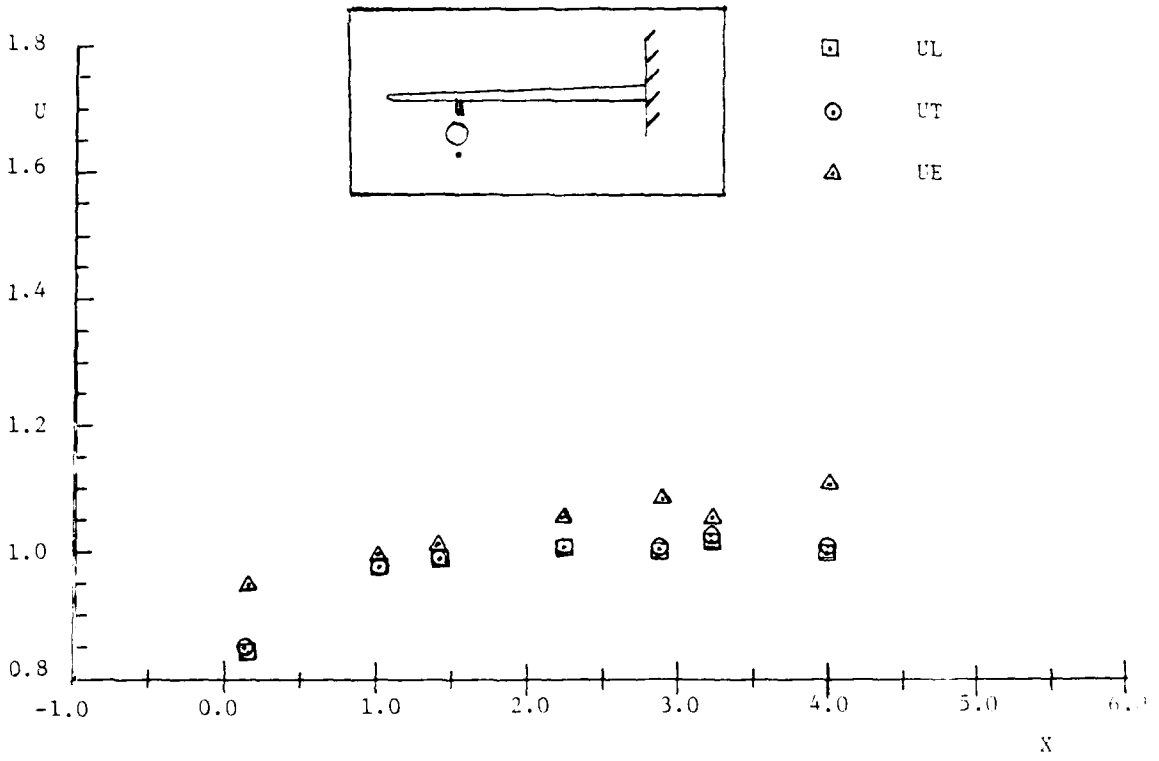


Figure 45. U Velocities at Y=0.0, Z=-0.525 for Configuration 121

M= 1.025, $\alpha=0^0$

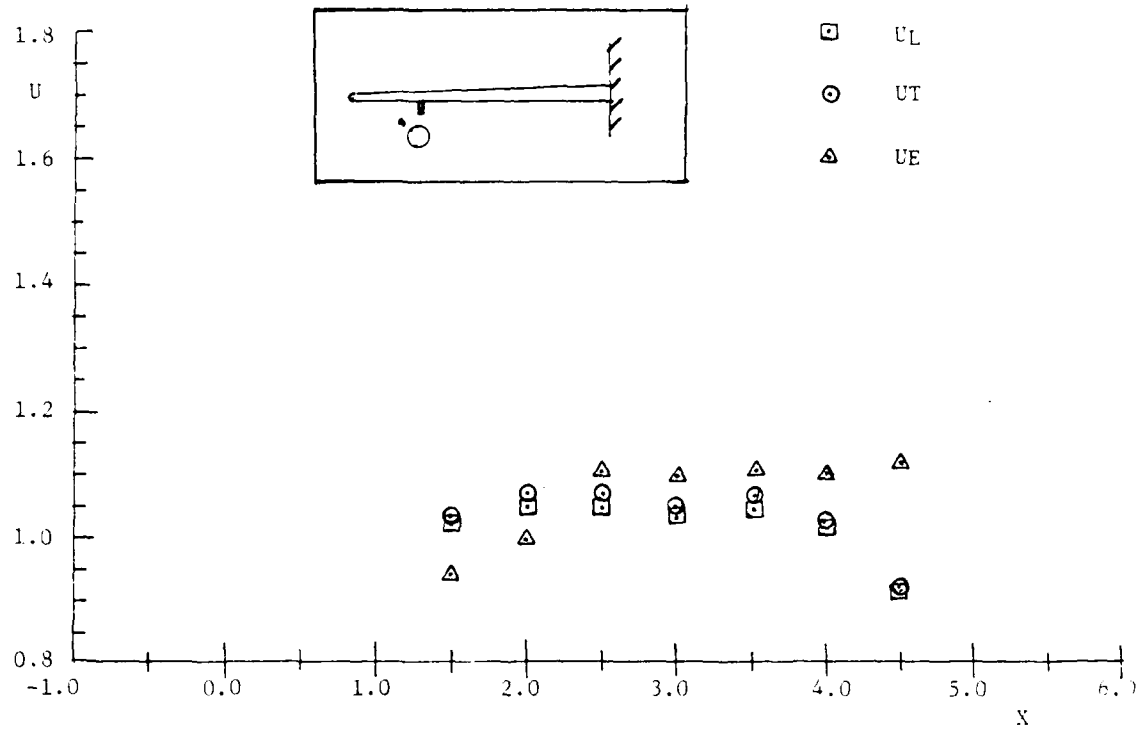


Figure 46. U Velocities at Y=-0.35, Z=0.392 for Configuration 121

M=1.025, $\alpha=0^0$

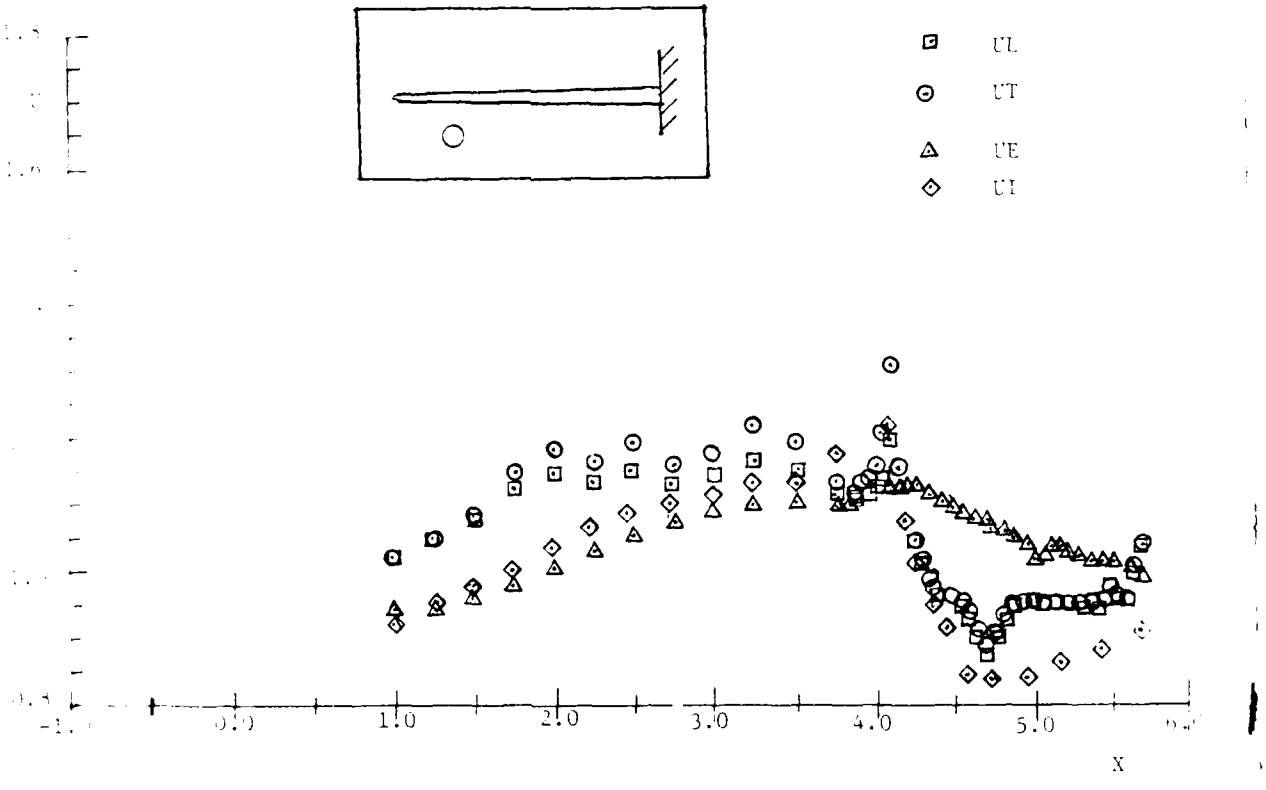


Figure 47. U Velocities in the Gap at Y=0.0, Z=0.392 for Configuration 101

M=0.975, $\alpha=0^0$

(With or Without Displacement Effect)

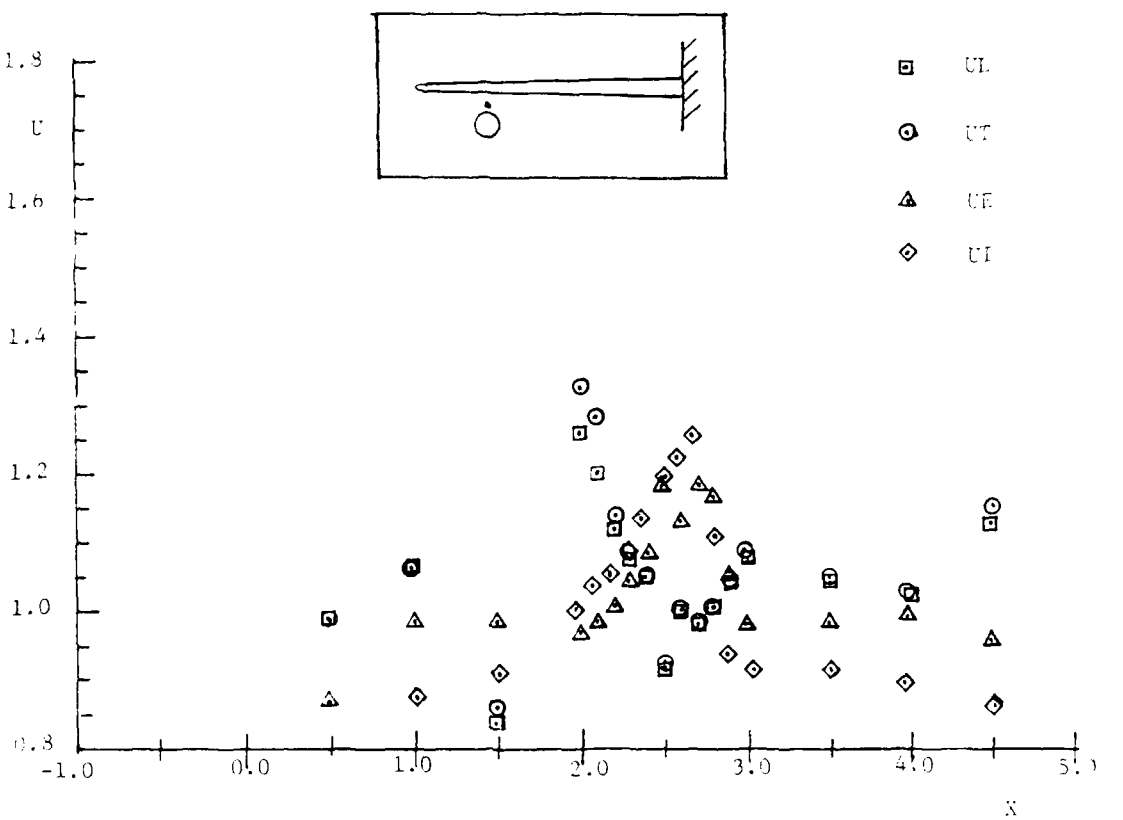


Figure 48. U Velocities in the Gap at Y=0.0, Z=0.442 for Configuration 102

M=0.92, $\alpha=0^0$

(Displacement Effect Included)

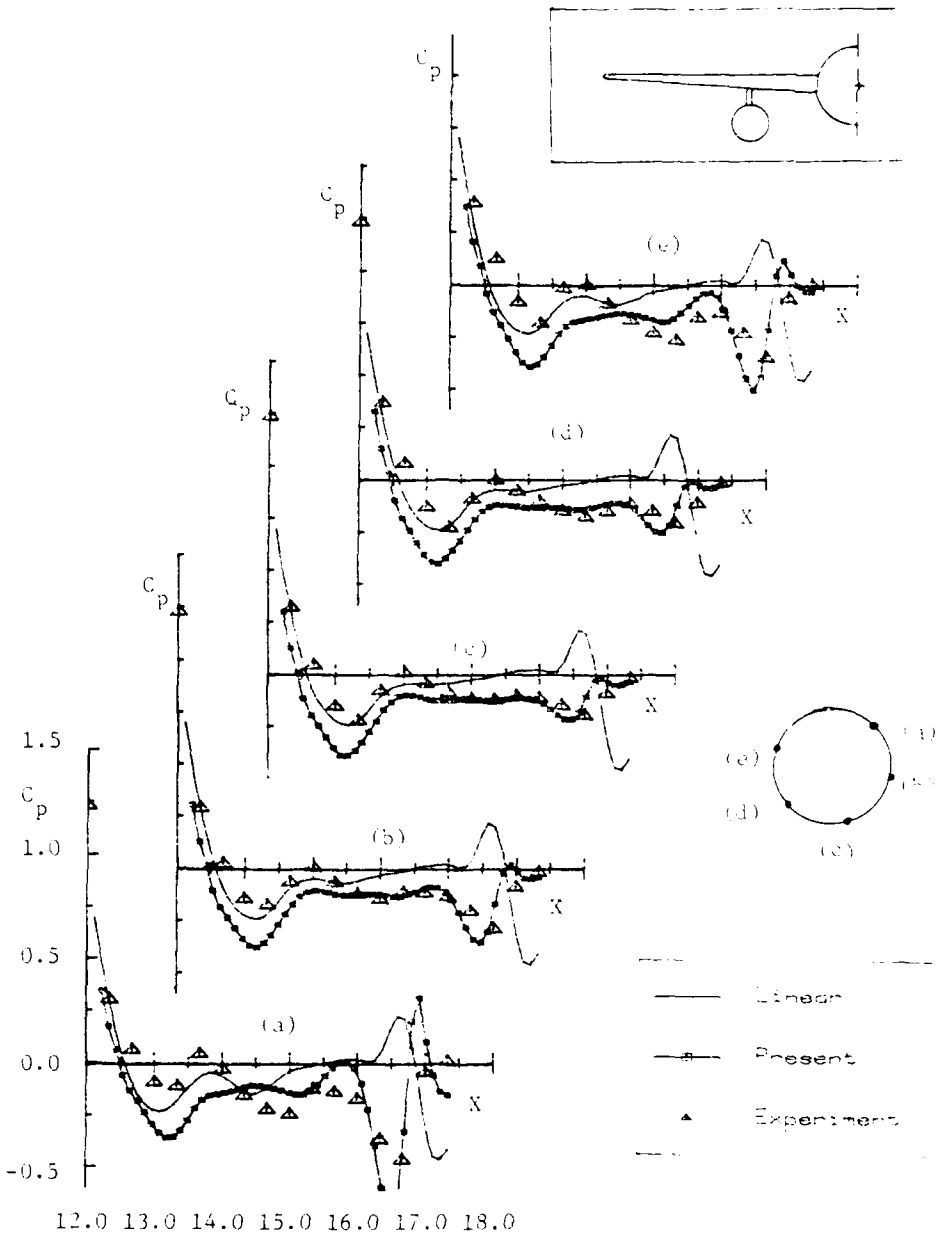


Figure 49. C_p on the Store Surface Along the Axis, M=0.925, =0°

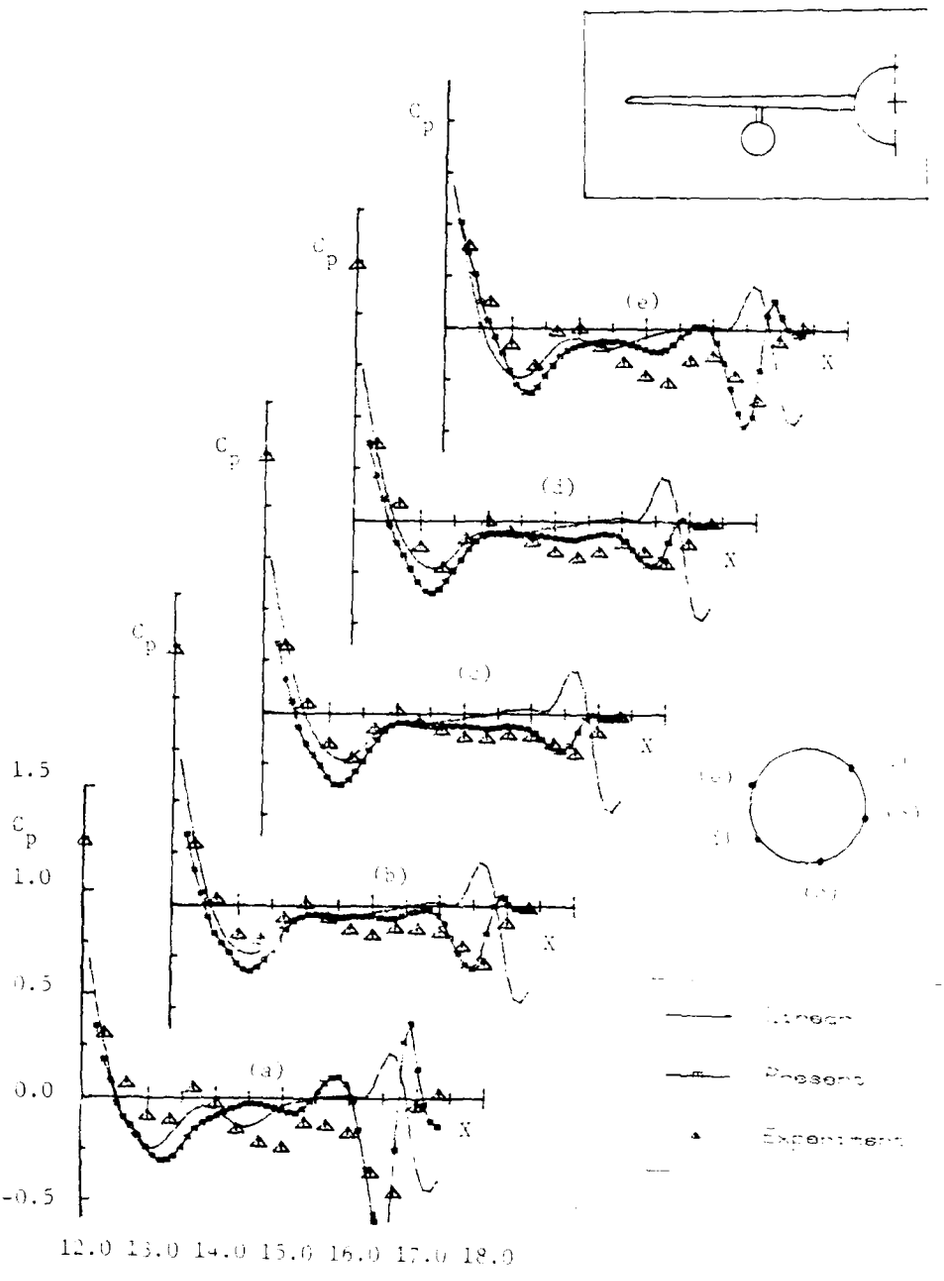


Figure 50. C_p on the Store Surface Along the Axis, M=0.925.

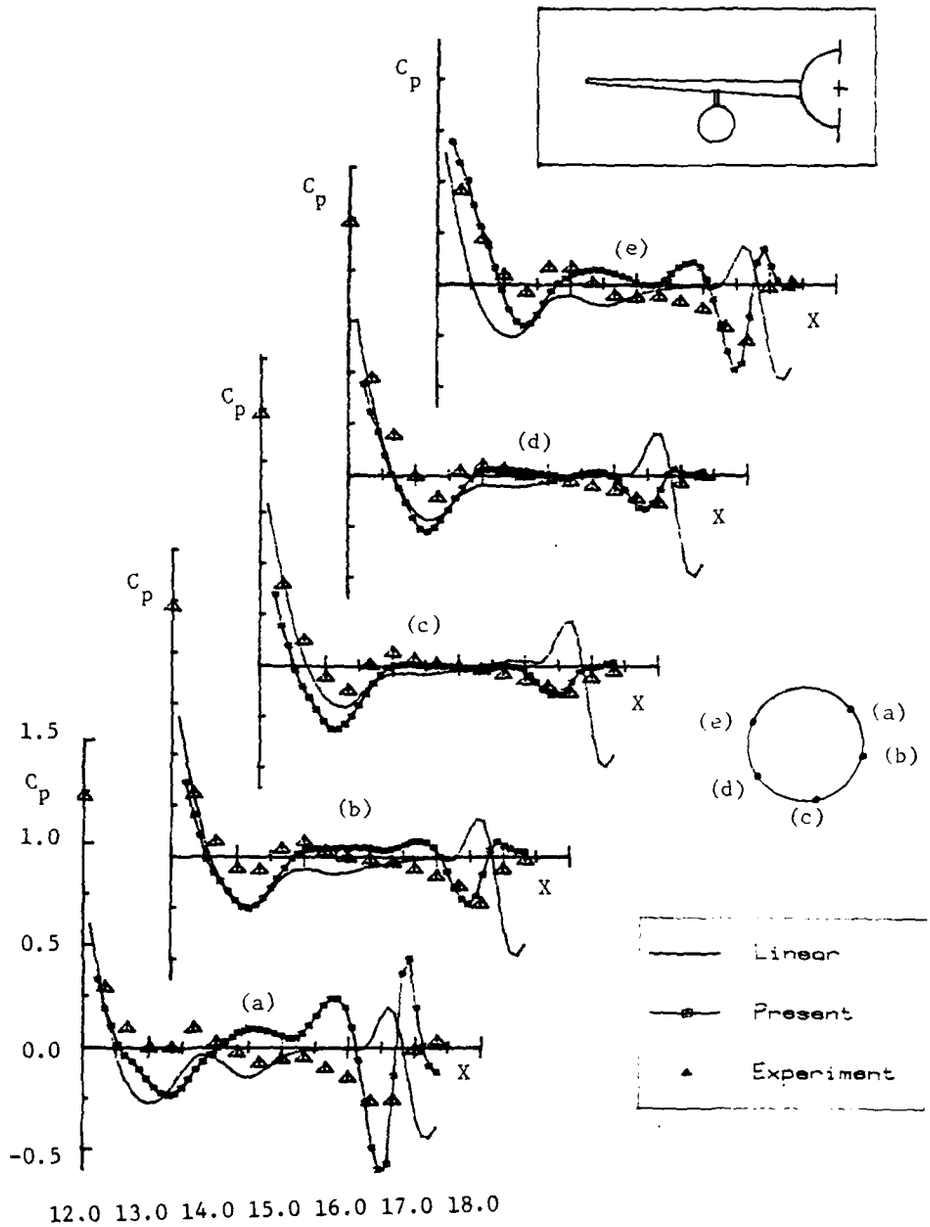


Figure 51. C_p on the Store Surface Along the Axis, M=0.925, $\alpha=5^o$

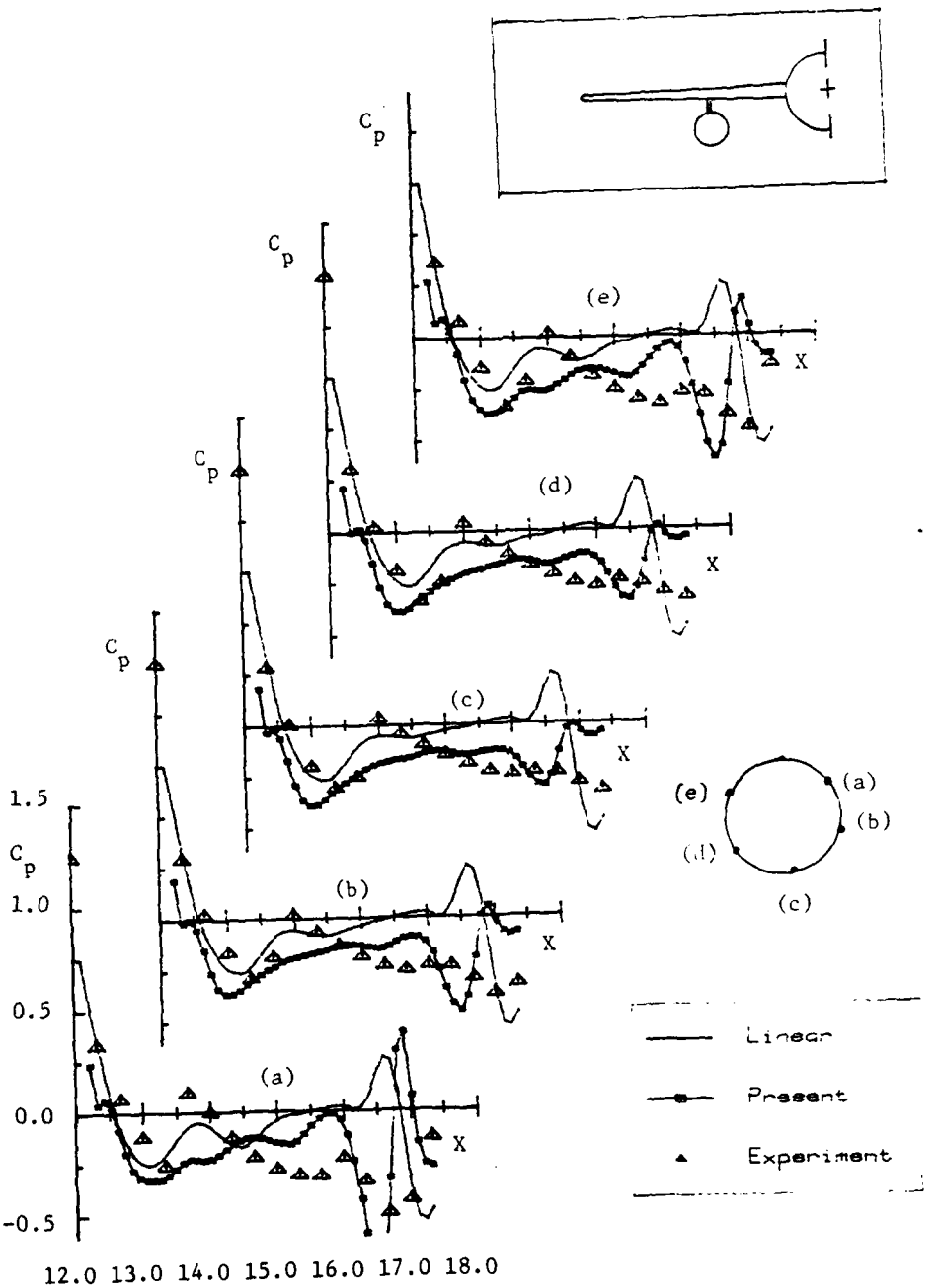


Figure 52. C_p on the Store Surface Along the Axis, M=0.952, $\alpha=0^o$

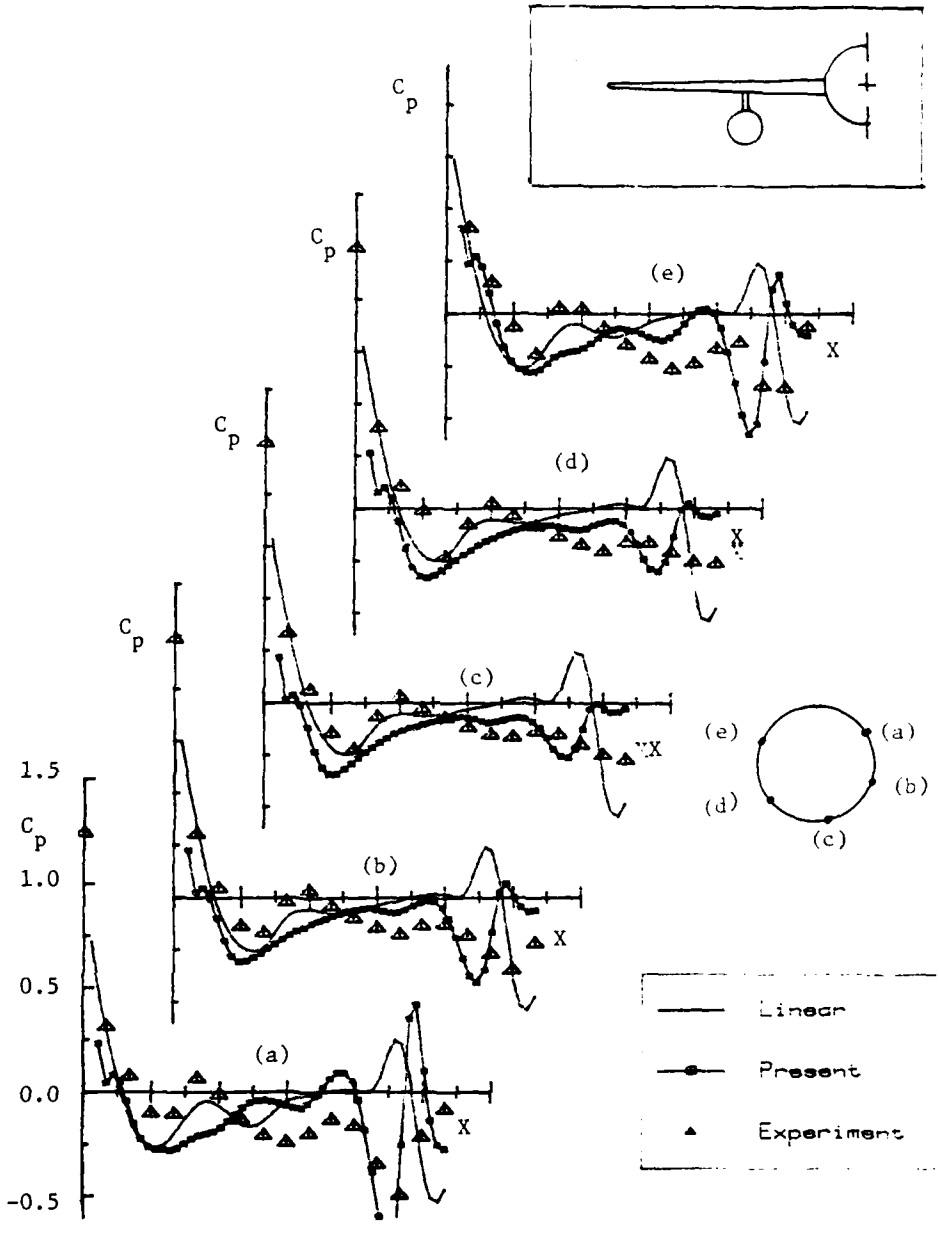


Figure 53. C_p on the Store Surface Along the Axis, M=0.952, $\alpha=2^0$

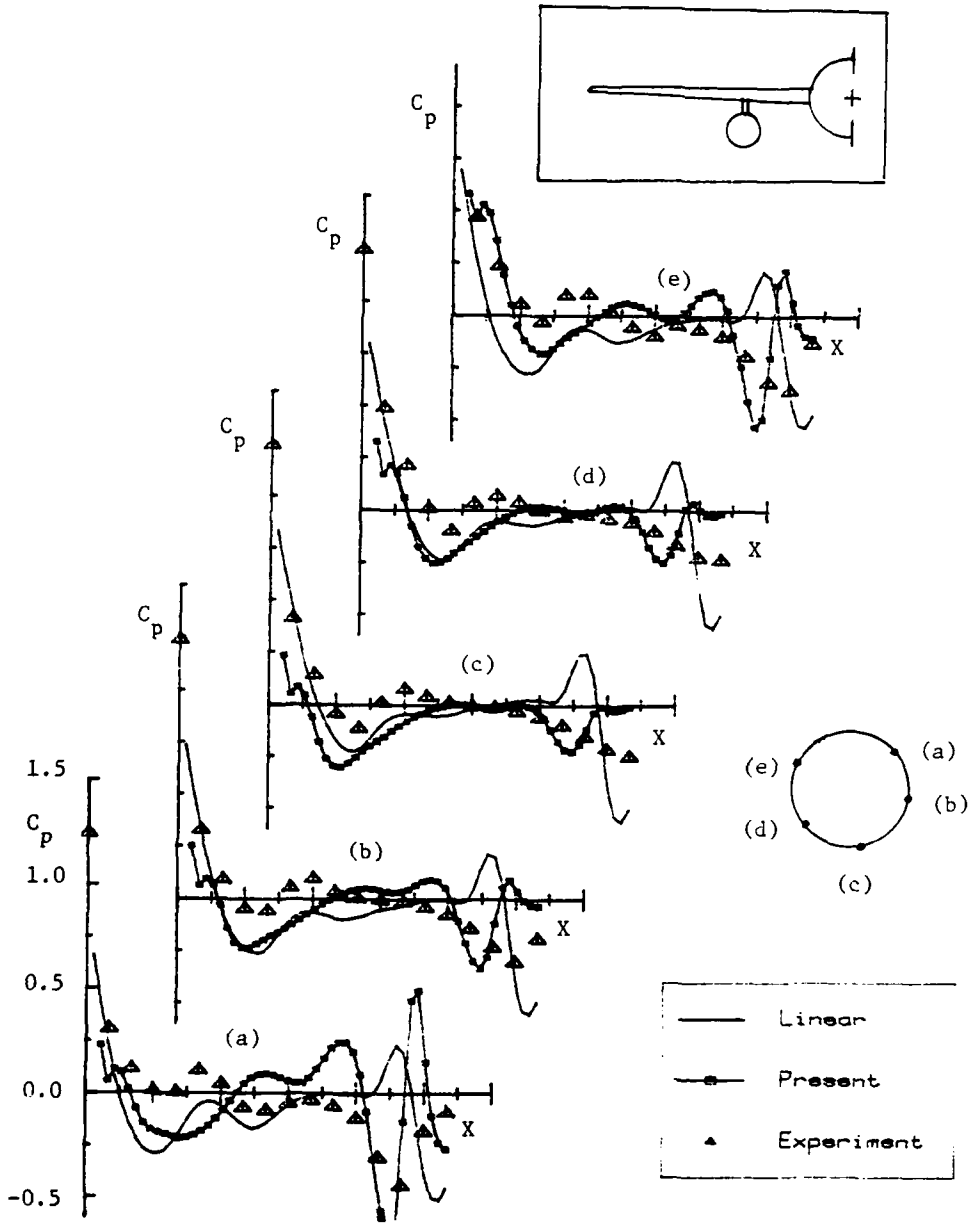


Figure 54. C_p on the Store Surface Along the Axis, M=0.952, $\alpha=5^o$

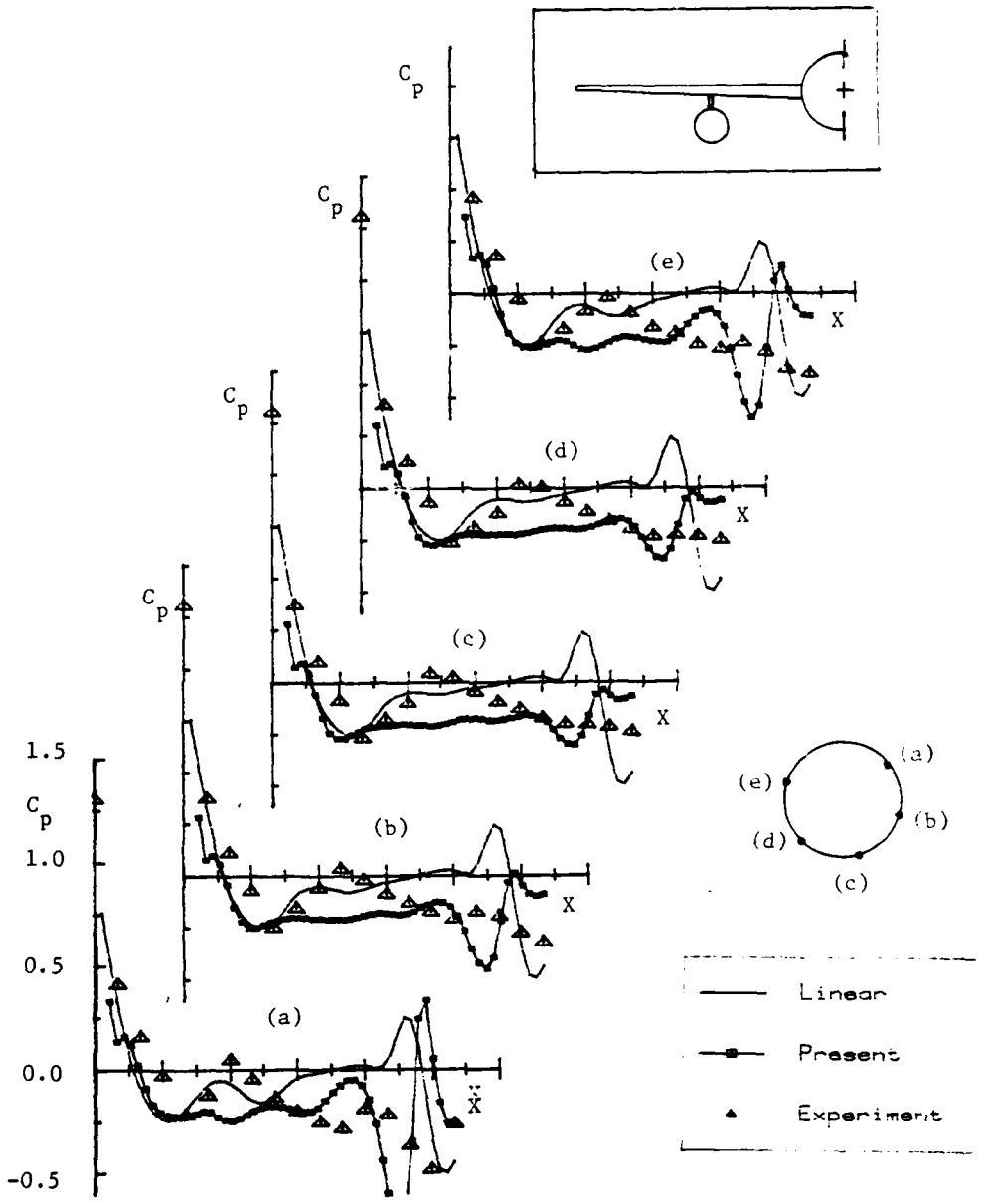


Figure 55. C_p on the Store Surface Along the Axis, M=1.052, $\alpha=0^o$

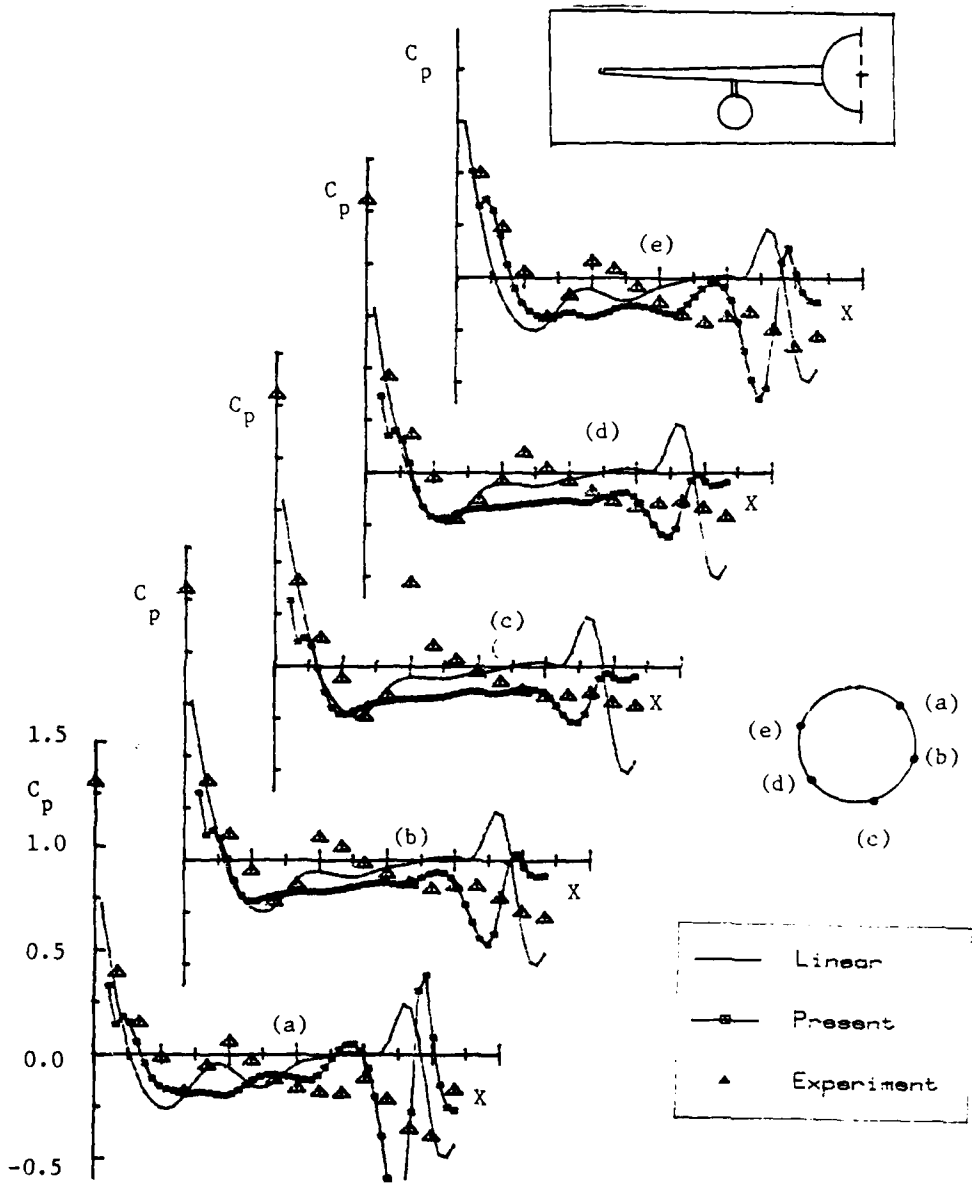


Figure 56. C_p on the Store Surface Along the Axis, M=1.052, $\alpha=2^{o}$

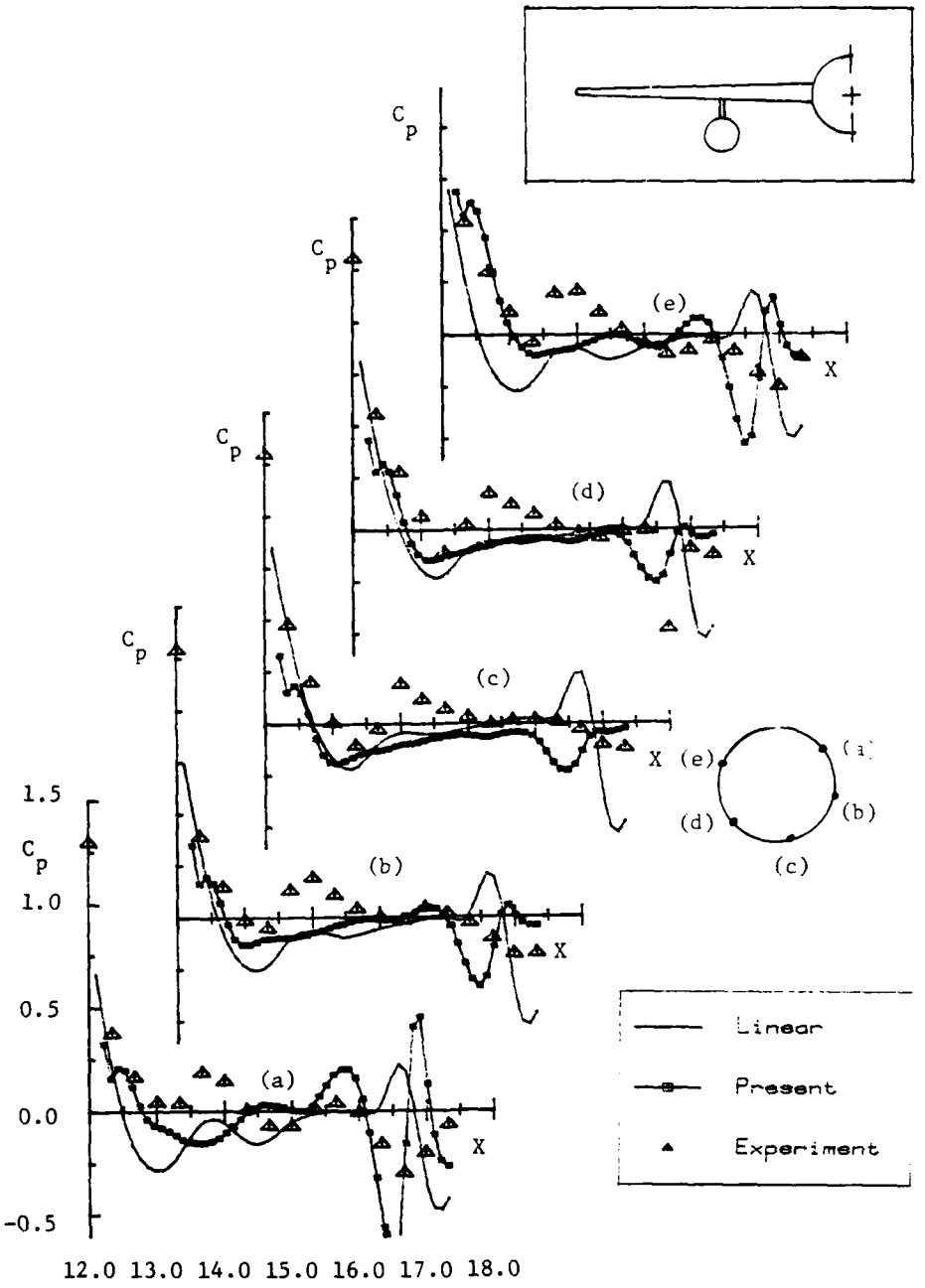
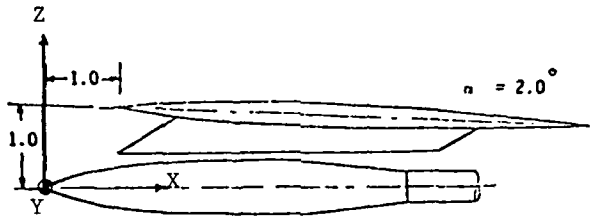


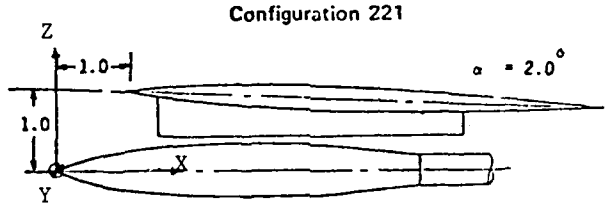
Figure 57. C_p on the Store Surface Along the Axis, M=1.052, $\alpha=5^{o}$

APPENDIX A

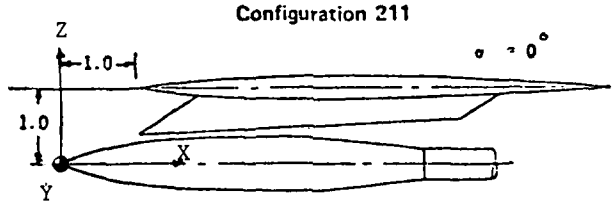
TEST CONFIGURATIONS



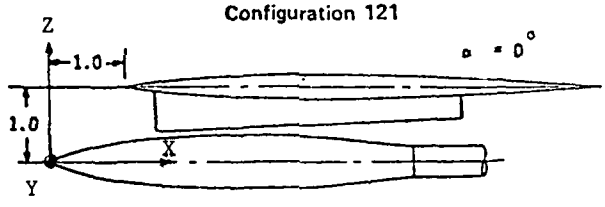
Configuration 221



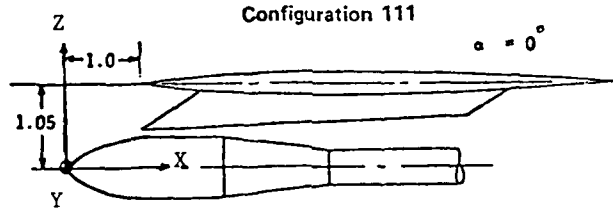
Configuration 211



Configuration 121



Configuration 111

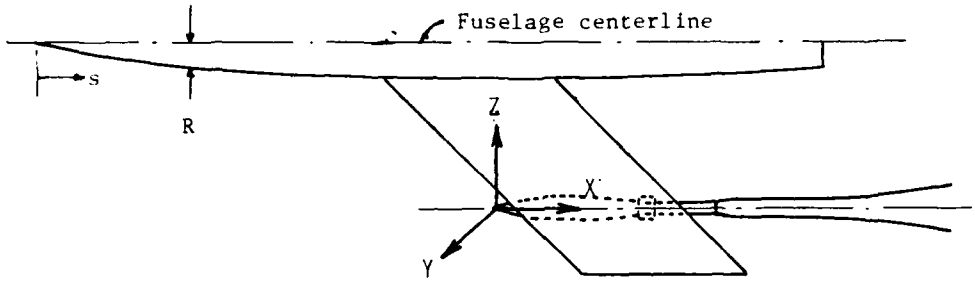


All Dimensions in Inches

Configuration 122

List of Configurations Investigated

Configurations (a)



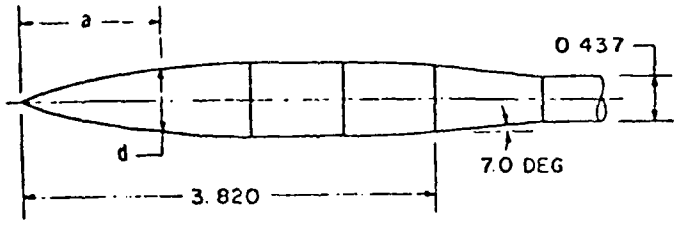
Configuration 001

All dimensions in inches

FUSELAGE CONTOUR

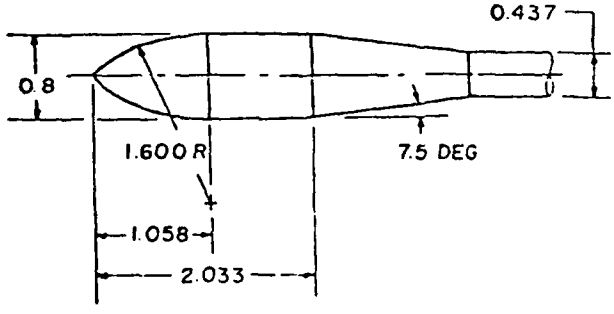
s	R
0	0
.9020	.351[illegible]
.8068	.4703
2.7115	.6628
3.6163	.82[illegible]
4.5183	.9625
5.4230	10725
6.3278	11578
7.2298	1.2183
8.1345	1.2458
8.800	1.2562
20.7215	1.2559
21.0898	1.2485
21.9945	1.2045
22.8965	1.1495
23.8013	1.0863
24.7060	1.0220
25.6180	.9598
26.5128	.8965
27.500	.8305

Configurations 2(a) (continued)



a	d	a	d	a	d
0.0000	0.0000	1.1834	0.5835	2.9833	.7000
0.2666	0.2313	1.3167	0.6093	3.1166	.6983
0.3834	0.3073	1.4500	0.6326	3.250	6934
0.5167	0.3777	1.5833	0.6534	3.3833	55
0.6500	0.4349	1.7166	0.6715	3.5166	0.6745
0.7834	0.4819	1.8500	0.6862	3.6500	0.6608
0.9167	0.5211	1.9833	0.6961	3.7833	0.6403
1.0500	0.5545	2.1166	0.7000	3.8200	0.6392

a. MK-83

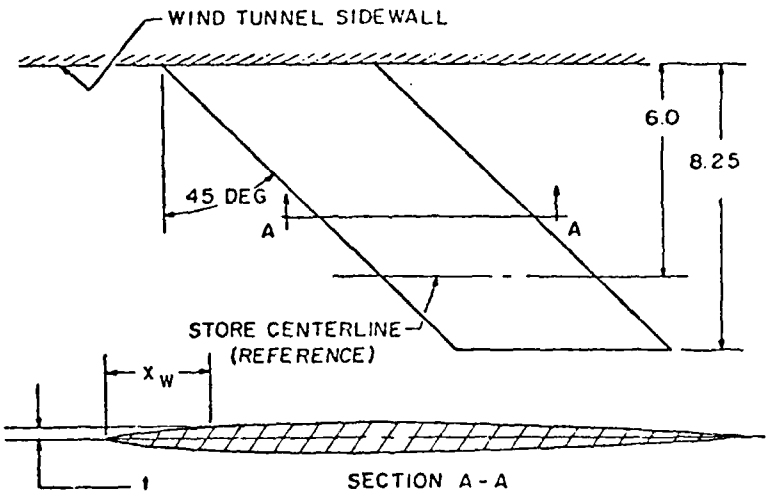


ALL DIMENSIONS IN INCHES

b. M-117

Detail and Dimensions of the Stores Used

Configurations (a) (continued)



x_w	t
0.0000	0.0000
0.0065	0.0095
0.0750	0.0284
0.1500	0.0420
0.3000	0.0623
0.4500	0.0783
0.6000	0.0914
0.9000	0.1124
1.2000	0.1274

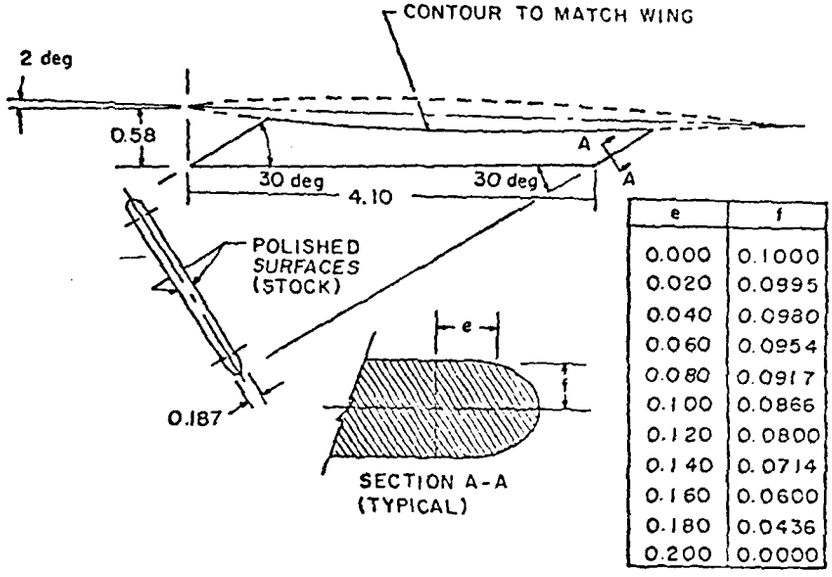
x_w	t
1.8000	0.1450
2.4000	0.1500
3.0000	0.1456
3.6000	0.1330
4.2000	0.1120
4.8000	0.0830
5.4000	0.0466
5.7000	0.0256
6.0000	0.0000

ALL DIMENSIONS IN INCHES

c. Wing

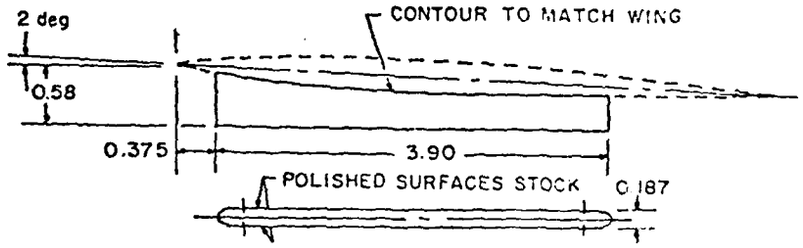
Wing Details and Dimensions

Configurations (a) (continued)



e	f
0.000	0.1000
0.020	0.0995
0.040	0.0980
0.060	0.0954
0.080	0.0917
0.100	0.0866
0.120	0.0800
0.140	0.0714
0.160	0.0600
0.180	0.0436
0.200	0.0000

d. Swept pylon

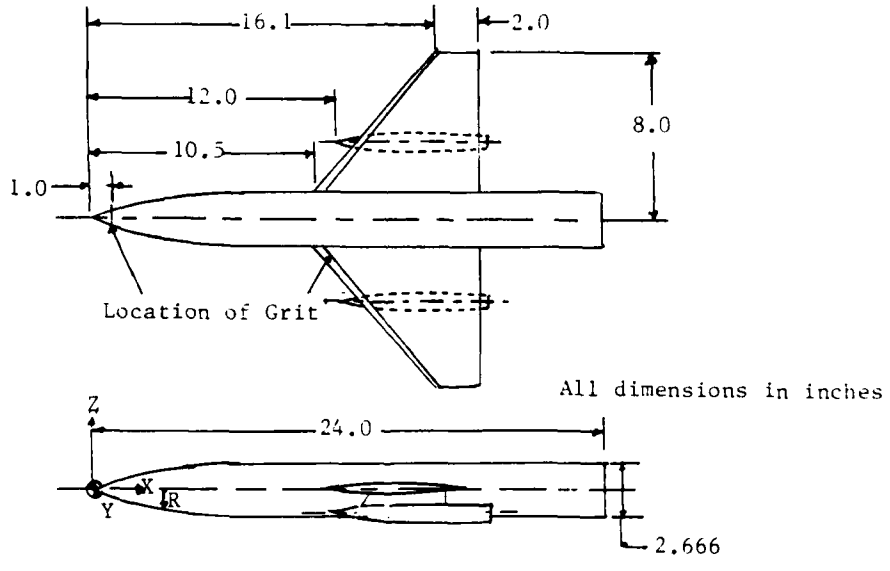


e. Unswept pylon

Pylon Details and Dimensions

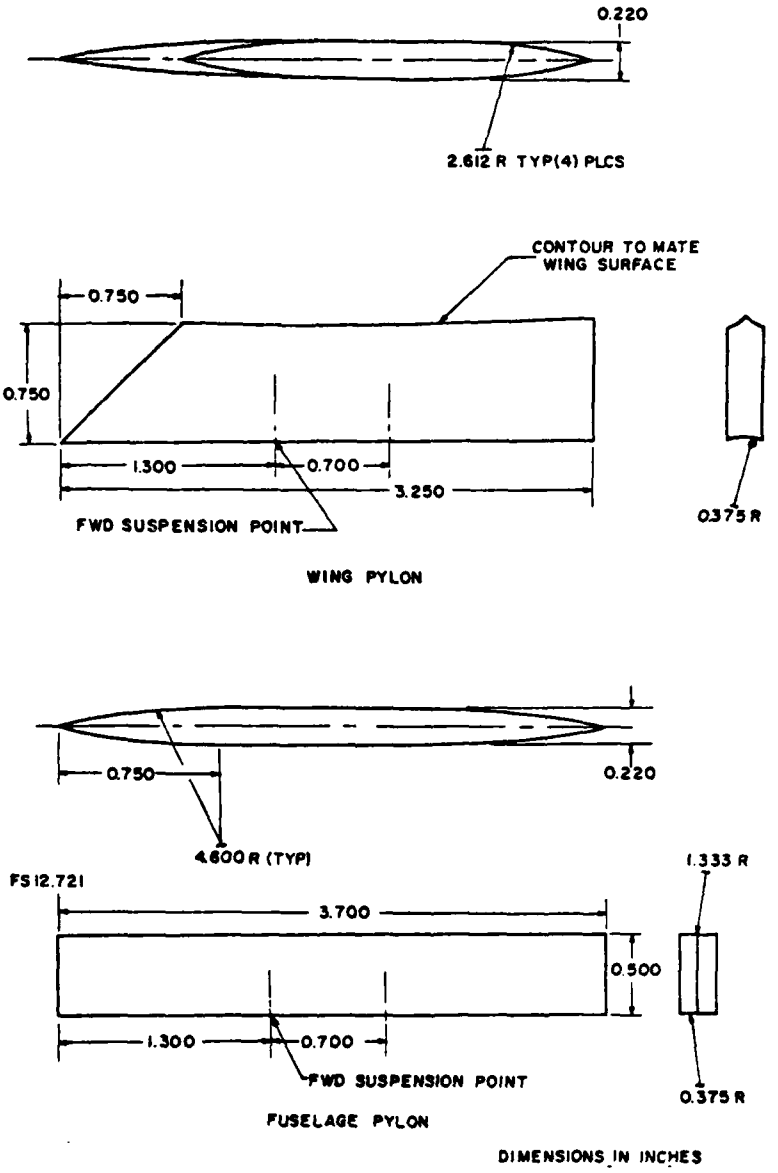
Configurations (a) (continued)

Nose Coordinates		4-percent Airfoil	
X, in.	R, in.	Percent Chord	t/c, percent
0.0	0.0	0.0	0.0
0.5	0.162	2.5	0.650
1.0	0.313	5.0	1.096
1.5	0.453	7.5	1.472
2.0	0.583	10.0	1.800
2.5	0.703	15.0	2.350
3.0	0 813	20.0	2.798
3.5	0.912	30.0	3.452
4.0	1.000	40.0	3.842
4.5	1.078	50.0	3.996
5.0	1.146	60.0	3.910
5.5	1.203	65.0	3.770
6.0	1.250	70.0	3.554
6.5	1.287	80.0	2.812
7.0	1.313	85.0	2.170
7.5	1.328	90.0	1.476
8.0	1.333	95.0	0.738
		100.0	0.0



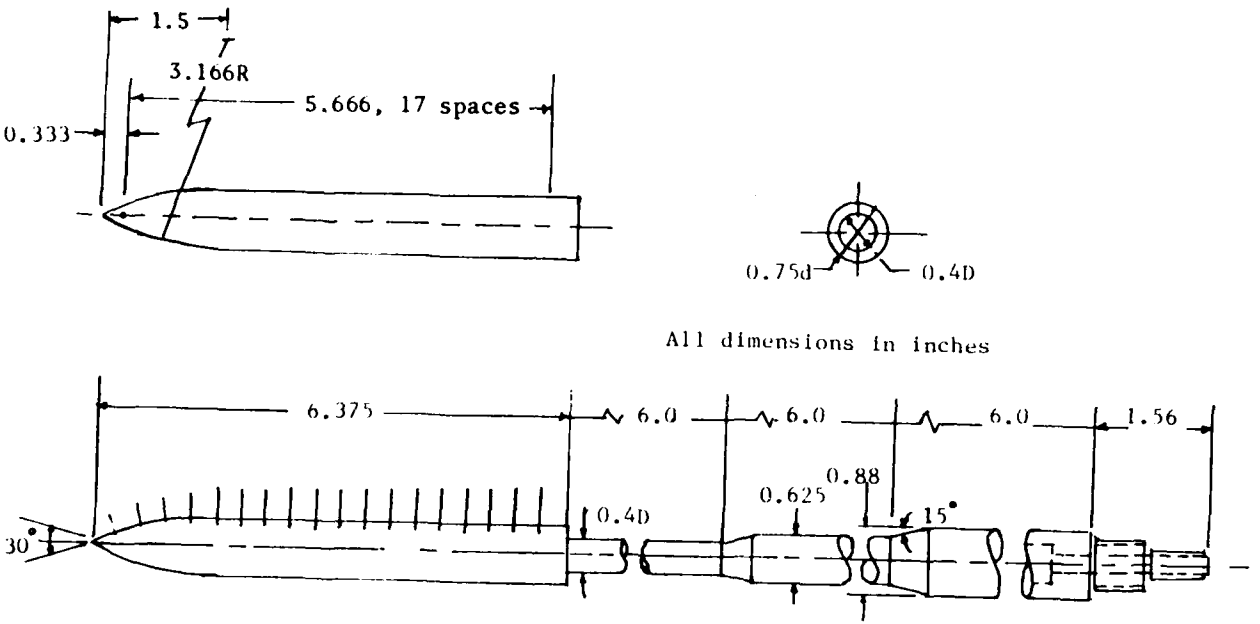
Configuration 24

Configurations (b)



Details and Dimensions of Pylons for Configuration 24

Configurations (b) (continued)



Details and Dimensions of Pressure Instrumented Store for Configuration 13

Configurations (b) (continued)

APPENDIX B

FORTRAN LISTING OF THE NON-LINEAR CORRECTION PROGRAM

```
C-----------------------------------------------------------------------
C         THIS IS THE PROGARM TO INCLUDE THE NON-LINEAR EFFECTS TO THE      .
C              LINEAR VELOCITIES OBTAINED FROM THE PANEL METHOD.            .
C-----------------------------------------------------------------------
          IMPLICIT REAL*8(A-H,O-Z)
         DIMENSION XDUMMY(300),YDUMMY(300),ZDUMMY(300),CPDUM(300)
          DIMENSION UDUMMY(300),VDUMMY(300),WDUMMY(300)
          DIMENSION X(600),Y(600),Z(600),U(600),V(600),W(600),CP(600)
          DIMENSION WBAR(600)
          DIMENSION UBAR(600),YBAR(600),ZBAR(600)
          DIMENSION VBAR(600)
          DIMENSION VSQLI(600),VSQNLI(600),CPLI(600),CPNLI(600)
          COMMON /POINT/ A,B,C
          COMMON /FIELD/ UDUMMY,VDUMMY,WDUMMY
           COMMON /VEL/UL,II
C******     M IS THE NUMBER OF FIELD POINTS FOR WHICH THE NONLINEAR CORRECTION
C******     IS SOUGHT.
C******     N IS THE NUMBER OF DUMMY POINTS USED IN INTEGRATION.PRESENT
C******     CASE IT IS 6*6*6 = 216 POINTS
C******     AMACH IS THE FREE STREAM MACH NUMBER
C******     NSTO IS FOR THE STORE INFORMATION.
C******  NSTO EQ 1 WHEN STORE IS PRESENT
C****** IF NSTO IS 1,THEN XSTO,YSTO,ZSTO HAS TO BE GIVEN IN ORDER
C****** TO CONVERT THE COORDINATES IN THE OUTPUT FROM WING
C****** COORDINATE SYSTEM TO STORE COORDINATE SYSTEM
C****** IF NSTO IS NOT 1,THEN  XSTO,YSTO,ZSTO DOES NOT HAVE TO BE GIVEN.
C******  VELOC IS THE FREE STREAM VELOCITY IN FT/SEC
C****** EITHER THIS CAN BE UNITY IN WHICH CASE THE OUTPUT VELOCITIES ARE
C****** U/U INFINITY. OR IT CAN BE THE ACTUAL VELOCITY AND IN THIS CASE
C****** THE OUTPUT  VELOCITY WILL BE IN FEET/SEC OR IN THE GIVEN
C****** UNIT AS DEFINED BY VELOC
C******  U,V,W ARE THE PERTURBATION VELOCITY COMPONENTS IN PHYSICAL COORDINATES
C******  UBAR ,VBAR,WBAR ARE THE REDUCED VELOCITY COMPONENTS
C******  X,Y,Z ARE THE POINTS IN PHYSICAL COORDIATE FOR WHICH THE VELOCITY
C****** DISTRIBUTION SOUGHT.
C****** XBAR,YBAR,ZBAR ARE THE REDUCED COORDINATES.FOR REFERENCE SEE NACA
C******  REPORT 1217
C******  UDUMMY ,VDUMMY ,WDUMMY ARE THE PERTURBATION VELOCITY COMPONENTS
C****** AT THE DUMMY POINTS IN THE SCALED SPACE FOR +INFINITY TO -INFINITY
C****** ATTACH THE OUTPUT OF HESSPROGRAM FOR THE DUMMY POINT AND FOR THE
C****** FIELD POINTS AFTER THE FIRST DATA IN THIS PROGRAM
          CALL ASSIGN (5,'DB0:NEW240NL.IN')
          CALL ASSIGN (7,'DB0:NEW240NL.OUT')
C           READ (5,345)M,AMACH,VELOC,NSTO,ALPSUN
         DATA M,AMACH,VELOC,NSTO,ALPSUN/86,0.925,1.,1,0.0/
C         DATA M,AMACH,VELOC,NSTO/86,0.925,1.0,0./
C           IF ( NSTO.EQ.1) READ (5,55)XSTO,YSTO,ZSTO
C         XSTO=5.
C         YSTO=6.
C         ZSTO=-1.
  55      FORMAT (3F10.6)
           N = 216
 345       FORMAT(1I5,2F10.6,I5,F10.6)
           READ (5,81)(X(I),Y(I),Z(I),U(I),V(I),W(I),CP(I),I=1,M)
 81        FORMAT(7F10.6)
```

```
      READ (5,81)(XDUMMY(I),YDUMMY(I),ZDUMMY(I),UDUMMY(I),
     1   VDUMMY(I),WDUMMY(I),CPDUM(I),I=1,N)
C       WRITE (6,342)
342   FORMAT(5X,'X',11X,'Y',11X,'Z',10X,'UL',6X,
     1     6X,'U',12X,'VL',11X,'V',11X,'WL',11X,'W',7X,'CPLIN',3X,'CPN')
      CONST=(AMACH*AMACH)*2.4/1.0
C     CONST=AMACH**1.1*2.4/1.0
      ANX=DCOS(ALPSUN*3.14159265/180.)
      ANZ=DSIN(ALPSUN*3.14159265/180.)
       WRITE (7,3285)
3285   FORMAT(5X,'THE CONFIGURATION IS 24 ,0.925')
      DO 3967 I=1,216
      UDUMMY(I)=UDUMMY(I)-ANX
      VDUMMY(I)=VDUMMY(I)
      WDUMMY(I)=WDUMMY(I)-ANZ
3967  CONTINUE
      DO 840 I=1,M
      U(I)=U(I)-ANX
      V(I)=V(I)
      W(I)=W(I)-ANZ
840   CONTINUE
       BETA=DSQRT(1.0-AMACH*AMACH)
       DO 2034 I=1,216
C      UDUMMY(I)=UDUMMY(I)*CONST/(BETA*BETA)
       UDUMMY(I)=UDUMMY(I)*CONST
2034   CONTINUE
     DO 10 I=1,M
      YBAR(I)=Y(I)*BETA
      ZBAR(I)=Z(I)*BETA
C      UBAR(I)=U(I)*CONST/(BETA*BETA)
  UBAR(I)=U(I)*CONST
C       PRINT 3490,U(I),UBAR(I)
3490   FORMAT(2F15.6)
C      VBAR(I)=V(I)*CONST/(BETA**3)
  VBAR(I)=V(I)*CONST/BETA
C      WBAR(I)=W(I)*CONST/(BETA**3)
  WBAR(I)=WBAR(I)*CONST/BETA
       A=X(I)
       B=YBAR(I)
       C=ZBAR(I)
       D=UBAR(I)
       CALL TRIPLE(E)
     II=1
C      G=UBAR(I)+(D*D)/2.D0-(0.079577)*E
 1000  G1=(2.0*UBAR(I)-1.0)
      G2=((2.0*0.079577*E)-(G1))
C     IF(UBAR(I).GT.1.0) G=(1.0+DSQRT((2.0*0.079577*E)-(G1)))
      IF(UBAR(I).GT.1.0.AND.G2.GT.0.0) G=(1.0+DSQRT(G2))
      IF(UBAR(I).LT.1.0.AND.G2.GT.0.0) G=(1.0-DSQRT(G2))
C     IF(UBAR(I).LT.1.0) G=(1.0-DSQRT((2.0*0.079577*E)-(G1)))
C     IF(UBAR(I).EQ.1.0) G=UBAR(I)
      IF(G2.EQ.0.0) G=1.0
      IF(G2.LT.0.0) G=UBAR(I)+(D*D)/2.0D0-(0.079577)*E
C     WA=DSQRT(3.D0)
C     G=(WA+1.0)*(1.0-(1.0-(WA-1.0)*UBAR(I))**0.5)
 1005  FORMAT(5X,2F15.8)
  UL=G
      CALL TRIPLE (E)
      II=II+1
      IF(II.GE.3) GO TO 1001
      GO TO 1000
 1001  CONTINUE
         CALL TRIP(SUM)
         VFIN=VBAR(I)-0.079577*SUM
         CALL TRIPE(TOT)
         WFIN=WBAR(I)-0.079577*TOT
C         TRANU=(G*BETA*BETA)/CONST
  TRANU=G/CONST
C         TRANV=(VFIN*BETA**3)/CONST
```

```
  TRANV=VFIN*BETA/CONST
C          TRANW=(WFIN*BETA**3)/CONST
  TRANW=WFIN*BETA/CONST
          TRANU=(TRANU+ANX)*VELOC
          TRANV=(Ø.Ø+TRANV)*VELOC
          TRANW=(ANZ+TRANW)*VELOC
          U(I)=(ANX+U(I))*VELOC
          V(I)=V(I)*VELOC
          W(I)=(ANZ+W(I))*VELOC
           V(I)=V(I)
           TRANV=TRANV
      VSQLI(I)=U(I)**2+V(I)**2+W(I)**2
      VSQNLI(I)=TRANU**2+TRANV**2+TRANW**2
          IF (NSTO.NE.1) GO TO 1ØØ
          X(I)=X(I)-XSTO
          Y(I)=-YSTO+Y(I)
          Z(I)=Z(I)-ZSTO
C1ØØ         WRITE (6,234)X(I),Y(I),Z(I),U(I),TRANU,V(I),TRANV,W(I),
 1ØØ      CONTINUE

C     1       TRANW
C            WRITE (6,345Ø)E
345Ø         FORMAT(1ØX,F15.6)
            WRITE (7,231)X(I),Y(I),Z(I),U(I),TRANU,V(I),TRANV,W(I),TRANW
231          FORMAT(6F1Ø.6)
234         FORMAT(4F12.6,5F12.6)
1Ø           CONTINUE
             STOP
             END
            SUBROUTINE TRIPLE(RESULT)
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION P(8),H(8)
       DIMENSION UDUMMY(3ØØ),VDUMMY(3ØØ),WDUMMY(3ØØ)
      COMMON /POINT/ XØ,YØ,ZØ
      COMMON /FIELD/ UDUMMY,VDUMMY,WDUMMY
      COMMON /VEL/ UL,II
C
      P(1)=Ø.932469514Ø3152DØ
      P(2)=-P(1)
      P(3)=Ø.6612Ø938646626 5DØ
      P(4)=-P(3)
      P(5)=Ø.2386191860 83197DØ
      P(6)=-P(5)
      H(1)=Ø.171324492379170DØ
      H(2)=H(1)
      H(3)=Ø.360761573Ø48139DØ
      H(4)=H(3)
      H(5)=Ø.46791393457269 1DØ
      H(6)=H(5)
      RESULT=Ø.DØ
      DO 5ØØ I=1,6
      DO 5ØØ J=1,6
      DO 5ØØ M=1,6
      K=((I-1)*36+(J-1)*6+M)
      IF(II.EQ.1)UL=UDUMMY(K)
      RESULT=RESULT+H(I)*H(J)*H(M)*F(P(I),P(J),P(M))
  5ØØ CONTINUE
      RETURN
      END
      DOUBLE PRECISION FUNCTION F(X,Y,Z)
C
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /POINT/ XØ,YØ,ZØ
      COMMON /VEL/ UL,II
C
      PI=3.14159265 4DØ
      PII=PI/2.ØDØ
      A=(XØ-DTAN(PII*X))**2
```

```
      B=(Y0-DTAN(PII*Y))**2
      C=(Z0-DTAN(PII*Z))**2
      D=(1.0/DCOS(PII*X))**2
      E=(1.0/DCOS(PII*Y))**2
      G=(1.0/DCOS(PII*Z))**2
      F=((PI**3)/8.)*((UL**2)/2.)*((2.0*A)-B-C)*(D*E*G)
     1 /((A+B+C)**2.5)
      RETURN
      END
           SUBROUTINE TRIP(RESULT)
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION P(8),H(8)
  DIMENSION UBAR(600)
      COMMON /POINT/ X0,Y0,Z0
  COMMON/VEL/UL,II
      DIMENSION UDUMMY(300),VDUMMY(300),WDUMMY(300)
      COMMON /FIELD/ UDUMMY,VDUMMY,WDUMMY
      P(1)=0.932469514203152D0
      P(2)=-P(1)
      P(3)=0.661209386466265D0
      P(4)=-P(3)
      P(5)=0.238619186083197D0
      P(6)=-P(5)
      H(1)=0.171324492379170D0
      H(2)=H(1)
      H(3)=0.360761573048139D0
      H(4)=H(3)
      H(5)=0.467913934572691D0
      H(6)=H(5)
      RESULT=0.D0
      DO 500 I=1,6
      DO 500 J=1,6
      DO 500 M=1,6
      K=((I-1)*36+(J-1)*6+M)
C      UL=UDUMMY(K)
      RESULT=RESULT+H(I)*H(J)*H(M)*FF(P(I),P(J),P(M))
  500 CONTINUE
      RETURN
      END
      DOUBLE PRECISION FUNCTION FF(X,Y,Z)
C
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /POINT/ X0,Y0,Z0
      COMMON /VEL/ UL,II
C
      PI=3.14159265 4D0
      PII=PI/2.0D0
      A=(X0-DTAN(PII*X))**2
      B=(Y0-DTAN(PII*Y))**2
      C=(Z0-DTAN(PII*Z))**2
      D=(1.0/DCOS(PII*X))**2
      E=(1.0/DCOS(PII*Y))**2
      G=(1.0/DCOS(PII*Z))**2
      FF=((PI**3)/8.)*((UL**2)/2.)*(3.0*DSQRT(A)*DSQRT(B))*(D*E*G)
     1 /((A+B+C)**2.5)
      RETURN
      END
           SUBROUTINE TRIPE(RESULT)
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION P(8),H(8)
  DIMENSION UBAR(600)
      COMMON /POINT/ X0,Y0,Z0
  COMMON /VEL/ UL,II
      DIMENSION UDUMMY(300),VDUMMY(300),WDUMMY(300)
      COMMON /FIELD/ UDUMMY,VDUMMY,WDUMMY
C
      P(1)=0.932469514203152D0
```

```
      P(2)=-P(1)
      P(3)=Ø.6612Ø938646626 5DØ
      P(4)=-P(3)
      P(5)=Ø.2386191860Ø83197DØ
      P(6)=-P(5)
      H(1)=Ø.171324492379170DØ
      H(2)=H(1)
      H(3)=Ø.360761573Ø48139DØ
      H(4)=H(3)
      H(5)=Ø.467913934572691DØ
      H(6)=H(5)
      RESULT=Ø.DØ
      DO 5ØØ I=1,6
      DO 5ØØ J=1,6
      DO 5ØØ M=1,6
      K=((I-1)*36+(J-1)*6+M)
C       UL=UDUMMY(K)
      RESULT=RESULT+H(I)*H(J)*H(M)*FFF(P(I),P(J),P(M))
 5ØØ CONTINUE
      RETURN
      END
      DOUBLE PRECISION FUNCTION FFF(X,Y,Z)
C
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /POINT/ XØ,YØ,ZØ
      COMMON /VEL/ UL,II
C
      PI=3.141592654DØ
      PII=PI/2.ØDØ
      A=(XØ-DTAN(PII*X))**2
      B=(YØ-DTAN(PII*Y))**2
      C=(ZØ-DTAN(PII*Z))**2
      D=(1.Ø/DCOS(PII*X))**2
      E=(1.Ø/DCOS(PII*Y))**2
      G=(1.Ø/DCOS(PII*Z))**2
      FFF=((PI**3)/8.)*((UL**2)/2.)*(3.Ø*DSQRT(A)*DSQRT(C))*(D*E*G)
     1 /((A+B+C)**2.5)
      RETURN
      END
```

```
C ------------------------------------------------------------------------
C       THE FOLLOWING TWO SUBROUTINES CALCULATES THE PRESSURE COEFFICIENTS  .
C            ON AND OFF THE BODY BASED ON THE SLENDER BODY ASSUMPTIONS.     .
C------------------------------------------------------------------------
      SUBROUTINE CPOFF(XC,YC,ZC,VXA,VYA,VZA)
      DIMENSION ALPHAX(10),ALPHAY(10),ALPHAZ(10)
      COMMON /OUTPUT/FPRINT,IANGLE
      COMMON /ANGLE/ MANGLE,ALPHAX,ALPHAY,ALPHAZ,SUNBET
      COMMON  / SCALE  / FC, BL
      AMACH=SQRT(1.-SUNBET*SUNBET)
      SUNBET=SQRT(1.-AMACH*AMACH)
      UA=VXA-ALPHAX(IANGLE)
      VA=VYA
      WA=VZA-ALPHAZ(IANGLE)
      VX=COS(ACOS(ALPHAX(IANGLE))/SUNBET)+(UA/(SUNBET*SUNBET))
      VY=VA/SUNBET
      VZ=SIN(ASIN(ALPHAZ(IANGLE))/SUNBET)+(WA/SUNBET)
      AG1=UA/(SUNBET*SUNBET)
      AG2=VA/SUNBET
      AG3=WA/SUNBET
      VSQ=VX**2+VY**2+VZ**2
      IF(SUNBET.EQ.1.) GO TO 3290
      CP=(1./(0.7*AMACH*AMACH))*((1.+(AMACH*AMACH/5.)*(1.-VSQ))**3.5
     1      -1.)
      GO TO 3300
 3290       CP=1.-VSQ
 3300       YC=YC/SUNBET*FC
      ZC=ZC/SUNBET*FC
      XC=XC*FC
      WRITE(7,540) XC,YC,ZC,VX,VY,VZ,CP
  540     FORMAT(7(2X,F10.6))
      RETURN
      END
      SUBROUTINE CPSUND(XC,YC,ZC,VXA,VYA,VZA)
      DIMENSION ALPHAX(10),ALPHAY(10),ALPHAZ(10)
      COMMON /ANGLE/ MANGLE,ALPHAX,ALPHAY,ALPHAZ,SUNBET
      COMMON /OUTPUT/ FPRINT,IANGLE
      COMMON  / SCALE  / FC, BL
      AMACH=SQRT(1.-SUNBET*SUNBET)
      UA=VXA-ALPHAX(IANGLE)
      VA=VYA
      WA=VZA-ALPHAZ(IANGLE)
      VX=COS(ACOS(ALPHAX(IANGLE))/SUNBET)+(UA/(SUNBET*SUNBET))
      VY=VA/SUNBET
      VZ=SIN(ASIN(ALPHAZ(IANGLE))/SUNBET)+(WA/SUNBET)
      AG1=UA/(SUNBET*SUNBET)
      AG2=VA/SUNBET
      AG3=WA/SUNBET
      VSQ=VX**2+VY**2+VZ**2
      IF(SUNBET.EQ.1.) GO TO 3290
      CP=(1./(0.7*AMACH*AMACH))*((1.+(AMACH*AMACH/5.)*(1.-VSQ))**3.5-1
     1      .)
      GO TO 3300
 3290       CP=1.-VSQ
 3300       YC=YC/SUNBET*FC
      ZC=ZC/SUNBET*FC
      XC=XC*FC
      WRITE(7,345) XC,YC,ZC,VX,VY,VZ,CP
  345     FORMAT(5X,7(F10.6,2X))
      RETURN
      END
```

APPENDIX C

FORTRAN LISTING OF SOME VISCOUS CORRECTION PROGRAMS

```
C-------------------------------------------------------------------------------
C   THIS PROGRAM READS AND REARRANGES THE INPUTS FROM THE CIRCUMFERENTIAL TO   .
C     AXIAL ORDER FOR NON-LIFTING SECTIONS. NOTE THAT A MAXIMUM OF THREE       .
C                              NON-LIFTING BODIES                              .
C          COULD BE PROCESSED. NOTE ALSO THAT A CORRECTION FOR THE WAKE        .
C            THICKNESS EFFECT HAS BEEN INCORPORATED BY ADDING 6*H AS           .
C               WAKE LENGTH, WHERE H IS THE SEMI BASE HEIGHT.                  .
C-------------------------------------------------------------------------------
      LOGICAL LASSTR1,NEWSTR1,OFF,SECT,LASSEC,FUSE,WALL,EVEN
      DIMENSION XLAST1(100),YLAST1(100),ZLAST1(100),
     1                  XLAST1P1(100),YLAST1P1(100),ZLAST1P1(100),
     2                 XLAST1P2(100),YLAST1P2(100),ZLAST1P2(100),
     3                 XOLD(20,80),YOLD(20,80),ZOLD(20,80),MLIN1(100),
     4                 MLIN1P(100),XNEW1B(10,100),YNEW1B(10,100),
     5                 ZNEW1B(10,100),XNEW1T(10,100),YNEW1T(10,100),
     6                 ZNEW1T(10,100),LNEW1B(10,100),MNEW1B(10,100),
     7                 LNEW1T(10,100),MNEW1T(10,100),XNEW1(1250),XNEW2(1250),
     8             YNEW1(1250),YNEW2(1250),ZNEW1(1250),ZNEW2(1250),
     9                 LNEW1(1250),LNEW2(1250),MNEW1(1250),MNEW2(1250),
     A             X1(650),Y1(650),Z1(650),X2(650),Y2(650),Z2(650),
     B             L1(650), M1(650),L2(650),M2(650),TITLE(17),
     C             X(1250),Y(1250),Z(1250),L(1250),M(1250),
     D                 NSORCE(10),NWAKE(10),NSTRIP(10),IPCV(10),IXFLAG(10),AL(10),
     E                 IG1(50,10),IGN(50,10),WIDTH(50,10),WIDKTP(2,10),NXT(50),
     F                 NSPLIT(10),NLINE1(10),NLINEN(10)
C
C       ASSIGN THE UNITS TO READ THE DATA.
C
      CALL ASSIGN(1,'DBA0:TEST1.IN')
      CALL ASSIGN(3,'DBA0:TEST1.OUT')
      CALL ASSIGN(5,'DBA0:TEMP.DAT')
       TYPE*,'PLEASE SAY 1 TO INDICATE THAT BL,FC,RE VALUES ARE INPUT'
      ACCEPT*,ISTOP
      IF(ISTOP.NE.1) STOP
C
C       FIRST READ AND WRITE DOWN THE TITLE AND THE CONTROL FLAGS.
C
      READ(1,1) TITLE
      WRITE(3,1) TITLE
1      FORMAT(17A4)
      READ(1,1234) CASE,LIFSEC,MOMENT,KUTTA,NOFF,LIST,MPR,IOUT,IG,LASWAK,
     1             IATACK,IWIDTH,ISAVE,SYM1,SYM2,AMACH,FC,BL
      WRITE(3,2) CASE,LIFSEC,MOMENT,KUTTA,NOFF,LIST,MPR,IOUT,IG,LASWAK,
     1             IATACK,IWIDTH,ISAVE,SYM1,SYM2,BL,FC,AMACH
2      FORMAT(A4,12I3,3X,3F4.2,F12.3,F5.3)
1234      FORMAT(A4,12I3,3F4.0,F12.0,F4.0)
      READ(1,3) (AL(I),I=1,IATACK)
      WRITE(3,3) (AL(I),I=1,IATACK)
3      FORMAT(6F10.6)
      IF(MOMENT.NE.0) THEN
```

```
      READ(1,4) ORIGNX,ORIGNY,ORIGNZ
      WRITE(3,4) ORIGNX,ORIGNY,ORIGNZ
      ELSE
      END IF
4      FORMAT(3F10.6)
      READ(1,1235) (NSORCE(I),NWAKE(I),NSTRIP(I),NSPLIT(I),NLINEI(I),
     1                NLINEN(I),IXFLAG(I),IPCV(I),I=1,LIFSEC)
      WRITE(3,5) (NSORCE(I),NWAKE(I),NSTRIP(I),IPCV(I),IXFLAG(I),I=1,LIFSEC)
5      FORMAT(4(3I4,2I2))
1235      FORMAT(2(7I4,I2))
      IF(IG.LE.0) GO TO 6
      DO 7 J=1,LIFSEC
      JSTRIP=NSTRIP(J)
      READ(1,8) (IG1(I,J),IGN(I,J),I=1,JSTRIP)
7      WRITE(3,8) (IG1(I,J),IGN(I,J),I=1,JSTRIP)
8      FORMAT(16I4)
6      IF(IWIDTH.EQ.0) GO TO 9999
      DO 800 J=1,LIFSEC
      JKK=0
      KX=IXFLAG(J)
      IF(KX.EQ.0) GO TO 9
      IF(KX-2) 1200,1300,1200
1200    JKX=1
      GO TO 9
1300      JKX=2
9      JS=JSTRIP-JKX
      JP1=JS+1
      READ(1,3) WIDKTR(1,J),(WIDTH(K,J),K=2,JP1),WIDKTR(2,J)
      WRITE(3,3) WIDKTR(1,J),(WIDTH(K,J),K=2,JP1),WIDKTR(2,J)
800      CONTINUE
9999      CONTINUE
C
C      NEXT  READ THE PANEL AND OFF BODY COORDINATES.
C
      I=0
7654      I=I+1
      READ(1,100)X1(I),Y1(I),Z1(I),L1(I),M1(I),X2(I),Y2(I),
     1                    Z2(I),L2(I),M2(I)
      IF(L1(I).EQ.3) THEN
      NONBDY=(I-1)*2+1
      ION=I
      GO TO 8765
      ELSE
      IF(L2(I).EQ.3) THEN
      NONBDY=I*2
      ION=I
      GO TO 8765
      ELSE
      GO TO 7654
      END IF
      END IF
8765      J=I
5432      J=J+1
      READ(1,201) X1(J),Y1(J),Z1(J),L1(J),X2(J),Y2(J),Z2(J),L2(J)
      IF(L1(J).EQ.3) THEN
      NOFBDY=(J-I-1)*2+1
      IOF=J
      GO TO 6543
      ELSE
      IF(L2(J).EQ.3) THEN
      NOFBDY=(J-I)*2
      IOF=J
      GO TO 6543
      ELSE
      GO TO 5432
      END IF
      END IF
100      FORMAT(2(3F10.0,2I1))
201        FORMAT(2(3F10.6,I1))
```

```
6543        TYPE*,'WALL SECTIONS ARE THOSE FOR WHICH WAKE EFFECTS ARE ABSENT'
        TYPE*,'SAY Ø IF THERE ARE NO WALLS'
        ACCEPT*,IWALL
        IF(IWALL.EQ.Ø) GO TO 9876
        TYPE*,'HOW MANY WALL SECTIONS?'
        ACCEPT*,NWALL
        IF(IWALL.EQ.1) WALL=.TRUE.
9876        IF(MOD(NONBDY,2).EQ.Ø) THEN
        N=NONBDY/2
        EVEN=.TRUE.
        ELSE
        N=(NONBDY+1)/2
        EVEN=.FALSE.
        END IF
C
C        NOW ARRANGE THE ONBODY DATA ALONE ONE COORDINATE PER LINE.
C
        REWIND 2
        DO 1Ø I=1,N
        WRITE(2) X1(I),Y1(I),Z1(I),L1(I),M1(I)
        IF(I.EQ.N.AND.EVEN.EQ..FALSE.) GO TO 1Ø
        WRITE(2) X2(I),Y2(I),Z2(I),L2(I),M2(I)
1Ø        CONTINUE
        LASSTR1=.FALSE.
        LASSEC=.FALSE.
        TYPE*,'ARE THERE FUSELAGE SECTION? SAY Ø IF NONE'
        ACCEPT*,IFUSE
        IF(IFUSE.EQ.Ø) FUSE= .FALSE.
        IF(IFUSE.NE.Ø) FUSE= .TRUE.
C
C        NEXT READ THE ARRANGED DATA TO COMPLY WITH THE SUBSEQUENT PATTERN.
C
        N=NONBDY
        REWIND 2
        DO 321Ø I=1,N
3210        READ (2) X(I),Y(I),Z(I),L(I),M(I)
        NEWSEC=Ø
        JJJ=1
        ITER=Ø
        NSUN1=Ø
        KWALCT=Ø
        LIFT=Ø
11111        CONTINUE
        K1=Ø
        DO 11Ø I=1,N
        IF(I.LT.JJJ) GO TO 11Ø
        IF(ITER.EQ.3) GO TO 112
        IF(L(I).EQ.2.AND.M(I).EQ.1) THEN
        GO TO 111
        ELSE
        END IF
        IF(L(I).EQ.2) SECT=.TRUE.
        IF(L(I).NE.2) SECT=.FALSE.
        IF(SECT) NEWSEC=NEWSEC+1
        IF(WALL.EQ..TRUE..AND.NEWSEC.LE.NWALL) NSUN1=NSUN1+1
        IF(NEWSEC.EQ.2.AND.ITER.EQ.2.AND.I.GT.JJJ) GO TO 111
        IF(NEWSEC.EQ.2.AND.ITER.EQ.1.AND.I.GT.JJJ) GO TO 2ØØØ
        IF(NEWSEC.EQ.2.AND.ITER.EQ.Ø) GO TO 1ØØØ
        NEWSEC=1
        IF(SECT) N1=Ø
        IF(L(I).GE.1) NEWSTR1= .TRUE.
        IF(L(I).EQ.Ø) NEWSTR1= .FALSE.
        IF(NEWSTR1) GO TO 1ØØ1
        GO TO 1ØØ2
1ØØ1        N1=N1+1
        IF(NEWSEC.EQ.1) NLINES1=N1
        GO TO 1ØØ3
1ØØ2        IF(NEWSTR1) MM1=Ø
        MM1=MM1+1
```

```
      IF(NEWSTR1.EQ..FALSE..AND.NEWSEC.EQ.1) MLINES1=MM1
      IF(SECT.EQ..TRUE..AND.NEWSEC.EQ.1) I0=I
      K=I-I0+1
      IF(LASSTR1.EQ..FALSE.) GO TO 1006
      K1=K1+1
      XLAST1(K1)=X(I)
      YLAST1(K1)=Y(I)
      ZLAST1(K1)=Z(I)
      IF(K1.GT.1.OR.ITER.EQ.1) GO TO 1009
      GO TO 1008
1003      IF(NEWSTR1.EQ..TRUE..AND.L(I+MM1).EQ.2.AND.NLINES1.GT.1)
     1                    LASSTR1=.TRUE.
      GO TO 1002
1006      IF(K.EQ.1) GO TO 1007
      GO TO 1010
1007       XLE1=X(I)
      YLE1=Y(I)
      ZLE1=Z(I)
      GO TO 1010
1008       BASEW=ABS(ZLAST1(K1)-ZLE1)
1009       XLAST1P1(K1)=XLAST1(K1)+0.3*6.*BASEW
      XLAST1P2(K1)=XLAST1P1(K1)+0.7*6.*BASEW
      YLAST1P1(K1)=YLAST1(K1)-0.3*(YLAST1(K1)-YLE1)
      YLAST1P2(K1)=YLE1
      ZLAST1P1(K1)=ZLAST1(K1)-0.3*(ZLAST1(K1)-ZLE1)
      ZLAST1P2(K1)=ZLE1
1010      CONTINUE
      XOLD(N1,MM1)=X(I)
      YOLD(N1,MM1)=Y(I)
      ZOLD(N1,MM1)=Z(I)
      GO TO 110
112      LIFT=LIFT+1
      WRITE(5)X(I),Y(I),Z(I),L(I),M(I)
110      CONTINUE
111      CONTINUE
      ITER=ITER+1
      LASSEC=.TRUE.
      IF(ITER.GT.3) GO TO 4321
C
C       NOW REARRANGE THE INPUTS
C
141      CONTINUE
      IF(WALL.EQ..TRUE..AND.NSUN1.LE.NWALL) NLINES1=NLINES1-2
      DO 101 N1=1,NLINES1+2
      NEWM1B=NLINES1+3-N1
      NEWM1T=NLINES1+1+N1
      IF(WALL.EQ..TRUE..AND.NSUN1.LE.NWALL) NLINES1=NLINES1+2
      IF(FUSE) GO TO 149
      J1=(MLINES1+1)/2
149      IF(FUSE.EQ..TRUE..AND.NEWSEC.LE.2) J1=MLINES1
      IF(ITER.EQ.1) NL1=J1
      IF(ITER.EQ.2) NL2=J1
      IF(ITER.EQ.3) NL3=J1
      DO 103 JJ=1,J1
      IF(FUSE) GO TO 150
      MLIN1(JJ)=(MLINES1+3)/2-JJ
      MLIN1P(JJ)=MLINES1+1-JJ
      MLIN1(J1)=MLIN1P(J1)
      GO TO 151
150      IF(NEWSEC.EQ.1) MLIN1P(JJ)=JJ
      IF(NEWSEC.EQ.2) MLIN1(JJ)=MLINES1-JJ+1
151      IF(N1.EQ.NLINES1+1) GO TO 1011
      IF(N1.EQ.NLINES1+2) GO TO 1012
      IF(FUSE.EQ..TRUE..AND.NEWSEC.EQ.1) GO TO 152
      XNEW1B(JJ,NEWM1B)=XOLD(N1,MLIN1(JJ))
      YNEW1B(JJ,NEWM1B)=YOLD(N1,MLIN1(JJ))
      ZNEW1B(JJ,NEWM1B)=ZOLD(N1,MLIN1(JJ))
      IF(FUSE) GO TO 102
152      XNEW1T(JJ,NEWM1T)=XOLD(N1,MLIN1P(JJ))
```

```
      YNEW1T(JJ,NEWM1T)=YOLD(N1,MLIN1P(JJ))
      ZNEW1T(JJ,NEWM1T)=ZOLD(N1,MLIN1P(JJ))
      GO TO 102
1011      IF(FUSE.EQ..TRUE..AND.NEWSEC.EQ.1) GO TO 153
      XNEW1B(JJ,NEWM1B)=XLAST1P1(MLIN1(JJ))
      YNEW1B(JJ,NEWM1B)=YLAST1P1(MLIN1(JJ))
      ZNEW1B(JJ,NEWM1B)=ZLAST1P1(MLIN1(JJ))
      IF(FUSE) GO TO 102
153       XNEW1T(JJ,NEWM1T)=XLAST1P1(MLIN1P(JJ))
      YNEW1T(JJ,NEWM1T)=YLAST1P1(MLIN1P(JJ))
      ZNEW1T(JJ,NEWM1T)=ZLAST1P1(MLIN1P(JJ))
      GO TO 102
1012      IF(FUSE.EQ..TRUE..AND.NEWSEC.EQ.1) GO TO 154
      XNEW1B(JJ,NEWM1B)=XLAST1P2(MLIN1(JJ))
      YNEW1B(JJ,NEWM1B)=YLAST1P2(MLIN1(JJ))
      ZNEW1B(JJ,NEWM1B)=ZLAST1P2(MLIN1(JJ))
      IF(FUSE) GO TO 102
154       XNEW1T(JJ,NEWM1T)=XLAST1P2(MLIN1P(JJ))
      YNEW1T(JJ,NEWM1T)=YLAST1P2(MLIN1P(JJ))
      ZNEW1T(JJ,NEWM1T)=ZLAST1P2(MLIN1P(JJ))
102       CONTINUE
103       CONTINUE
101       CONTINUE
      IF(FUSE.EQ..TRUE..AND.NEWSEC.EQ.1) GO TO 11111
      IF(FUSE.EQ..TRUE..AND.NEWSEC.EQ.2) FUSE= .FALSE.
      GO TO 121
1000      NEWSEC=1
       ITER=1
       LASSTR1=.FALSE.
      JJJ=JJJ+NLINES1*MLINES1
      REWIND 5
      GO TO 141
2000      NEWSEC=2
      LASSTR1=.FALSE.
      JJJ=JJJ+NLINES1*MLINES1
      ITER=ITER+1
      GO TO 141
121       CONTINUE
      DO 130 I=1,J1
      IF(WALL.EQ..TRUE..AND.NSUN1.LE.NWALL) NLINES1=NLINES1-2
      DO 140 J=1,(2*NLINES1+3)
      IF(WALL.EQ..TRUE..AND.NSUN1.LE.NWALL) NLINES1=NLINES1+2
      IF(J.EQ.1) LNEW1B(I,J)=1
      IF(J.EQ.1.AND.I.EQ.1) LNEW1B(I,J)=2
      IF(J.GT.NLINES1+2) GO TO 160
      WRITE(5)XNEW1B(I,J),YNEW1B(I,J),ZNEW1B(I,J),LNEW1B(I,J)
     1               ,MNEW1B(I,J)
      GO TO 140
160       WRITE(5)XNEW1T(I,J),YNEW1T(I,J),ZNEW1T(I,J),LNEW1T(I,J)
     1               ,MNEW1T(I,J)
140       CONTINUE
130       CONTINUE
C
C       FOR MULTIPLE SECTIONS
C
      IF(IWALL.EQ.1.AND.NWALL.EQ.NSUN1) WALL=.FALSE.
      IF(LASSEC.EQ..FALSE.) GO TO 11111
      JJJ=JJJ+NLINES1*MLINES1
      IF(ITER.EQ.3) GO TO 11111
4321      IF(IFUSE.EQ.1.OR.NWALL.EQ.1) IWAKE=4*(NL2+NL3)
      IF(IFUSE.EQ.0.AND.IWALL.EQ.0) IWAKE=4*(NL1+NL2+NL3)
      IF(NWALL.EQ.2) IWAKE=4*(NL3)
      NN=JJJ+IWAKE+LIFT
      II=MOD(NN,2)
      IF(II.NE.0) NN=NN+1
C
C       NEXT REWRITE TWO PANELS PER LINE
C
      REWIND 5
```

```
      DO 40 I=1,NN/2
      READ(5) XNEW1(I),YNEW1(I),ZNEW1(I),LNEW1(I),MNEW1(I)
      IF(II.NE.0.AND.I.EQ.NN/2) GO TO 40
      READ(5)XNEW2(I),YNEW2(I),ZNEW2(I),LNEW2(I),MNEW2(I)
40       CONTINUE
      WRITE(3,200) (XNEW1(I),YNEW1(I),ZNEW1(I),LNEW1(I),MNEW1(I),
     1          XNEW2(I),YNEW2(I),ZNEW2(I),LNEW2(I),MNEW2(I),I=1,NN/2)
194      CONTINUE
123      CONTINUE
200      FORMAT(2(3F10.6,2I1))
C
C       WRITE DOWN THE OFF BODY POINTS IN THE SAME OUTPUT FILE.
C
      WRITE(3,201) (X1(I),Y1(I),Z1(I),L1(I),X2(I),Y2(I),Z2(I),L2(I),
     1                    I=ION+1,IOF)
C
C       FINALLY READ AND WRITE THE BOUNDARY LAYER PARAMETERS.
C
      READ(1,22222) UI,RI,ISM,IBETA,NOCAL,NTRANS
      WRITE(3,22222) UI,RI,ISM,IBETA,NOCAL,NTRANS
22222       FORMAT(2F15.6,3I5,I2)
      READ(1,33333) (NXT(I),I=1,NTRANS)
      WRITE(3,33333) (NXT(I),I=1,NTRANS)
33333       FORMAT(I5)
      STOP
      END
```

```
C*DECK NOLIFT
C      LIST,NONE
       SUBROUTINE NOLIFT
C
C*****THIS SUBPROGRAM COMPUTES THE GEOMETRIC QUANTITIES OF THE
C*****NON-LIFTING ELEMENTS. REFER TO REPORT NO. E.S. 40622.
C
          REAL          NX ,NY ,NZ,  IXX,  IXY,  IYY
       DIMENSION     XIN ( 4 )      ,     YIN ( 4 )
       DIMENSION     ZIN ( 4 )      ,     XA ( 500 )
       DIMENSION     YA ( 500 )     ,     ZA ( 500 )
       DIMENSION     XB ( 500 )     ,     YB ( 500 )
       DIMENSION     ZB ( 500 )     ,     XI ( 4 )
       DIMENSION     ETA ( 4 )      ,     TITLE ( 18 )
       DIMENSION     XC ( 1100 )    ,     YC  ( 1100 )
       DIMENSION     ZC ( 1100 )    ,     XN  ( 1100 )
       DIMENSION     YN ( 1100 )    ,     ZN  ( 1100 )
       DIMENSION     UNITVX ( 1100),      UNITVY ( 1100 )
       DIMENSION     UNITVZ ( 1100),      A   ( 1100 )
       DIMENSION NSORNL(50)
C
C                          ---------- * ----------
C
       COMMON  / POINTS / XA, YA, ZA, XB, YB, ZB
       COMMON  / CONFLG / TITLE, CASE, LIFSEC, LIST, NOFF, IOUT, KONTRL,
      1                   KUTTA, MPR
       COMMON  /  MISC  / N, LABEL, NPRT, LPRT, MMIN
       COMMON  / IADDRS / ISAVE, JJ, ICOUNT
```

```
      COMMON  / CTABLE / XC, YC, ZC ,AMACH
      COMMON  / NORMAL / XN, YN, ZN
      COMMON  / ATABLE / A
      COMMON  / UVETOR / UNITVX, UNITVY, UNITVZ
      COMMON  / SCALE  / FC, BL
      DATA BTYPE / 4HNLIF /
C                          ---------- * ----------
C
CT5ØØ FORMAT ( 1HØ, 14X, 5HN   M, 7X, 4( 1HX, 11X ), 2HNX, 1ØX, 5H  XØ ,
CT    1          1ØX, 1HD, 1ØX, 7HTYPE OF / 27X, 4( 1HY, 11X ), 2HNY, 1ØX,
CT    2          5H  YØ , 1ØX, 1HT, 1ØX, 7HELEMENT / 27X, 4( 1HZ, 11X ),
CT    3          2HNZ, 1ØX, 5H  ZØ , 1ØX, 1HA / 14X, 6H------, 4( 3X,
CT    4          9H--------- ), 3X, 1Ø( 1H- ), 4X, 9( 1H- ), 4X, 1Ø( 1H- :
CT    5           , 5X, 7( 1H- ) / )
CT51Ø FORMAT ( 1HØ, 15X, I4, 4F12.6, 2F13.6, 1PE14.4 /
CT    1       ( 2ØX, ØP4F12.6, 2F13.6, 1PE14.4 ) )
CT52Ø FORMAT ( 1HØ, 11X, 2I4, 4F12.6, 2F13.6, 1PE14.4, 6X, A4 / ( 2ØX,
CT    1          ØP4F12.6, 2F13.6, 1PE14.4 ) )
CT53Ø FORMAT ( 1HØ, 53X, 13(1H-), 3H * , 14(1H-) )
 5ØØ  FORMAT(12X,4HN  M,5X,2HXØ,8X,2HYØ,8X,2HZØ,8X,2HXN,8X,2HYN,
     1       8X,2HZN,9X,1HD,9X,1HT,9X,1HA,6X,4HTYPE)
 5Ø1  FORMAT(1ØX,2I3,9F1Ø.6,3X,A6)
 5Ø2  FORMAT(13X,I3,9F1Ø.6)
C
C                          ---------- * ----------
C
      SUNBET=SQRT(1.-AMACH*AMACH)
      WRITE ( 4 ) LABEL
      DO 2ØØØ I = 1, MMIN
      KONTRL = KONTRL + 1
      XIN (1) = XA (I)
      XIN (2) = XB (I)
      XIN (3) = XB (I+1)
      XIN (4) = XA (I+1)
      YIN (1) = YA (I)
      YIN (2) = YB (I)
      YIN (3) = YB (I+1)
      YIN (4) = YA (I+1)
      ZIN (1) = ZA (I)
      ZIN (2) = ZB (I)
      ZIN (3) = ZB (I+1)
      ZIN (4) = ZA (I+1)
      DO 6ØØ K = 1, 4
      XIN ( K ) = XIN ( K ) / FC
      YIN ( K ) = YIN ( K ) / FC
      ZIN ( K ) = ZIN ( K ) / FC
  6ØØ CONTINUE
C
C                          ---------- * ----------
C
C
C*****FORM DIAGONAL VECTORS, EQUATION ( 64 ).
C
      T1X = XIN (3) - XIN (1)
      T2X = XIN (4) - XIN (2)
      T1Y = YIN (3) - YIN (1)
      T2Y = YIN (4) - YIN (2)
      T1Z = ZIN (3) - ZIN (1)
      T2Z = ZIN (4) - ZIN (2)
C
C*****FORM CROSS PRODUCT N = T2 X T1, EQUATION ( 65 ).
C
      NX = T2Y * T1Z - T1Y * T2Z
      NY = T1X * T2Z - T2X * T1Z
      NZ = T2X * T1Y - T1X * T2Y
      VN = SQRT ( NX * NX + NY * NY + NZ * NZ )
C
C*****FORM UNIT NORMAL VECTOR, EQUATION ( 66 ).
C
```

```
      NX = NX / VN
      NY = NY / VN
      NZ = NZ / VN
C
C*****COMPUTE AVERAGE POINT, EQUATION ( 68 ).
C
      AVX = Ø.25 * ( XIN(1) + XIN(2) + XIN(3) + XIN(4) )
      AVY = Ø.25 * ( YIN(1) + YIN(2) + YIN(3) + YIN(4) )
      AVZ = Ø.25 * ( ZIN(1) + ZIN(2) + ZIN(3) + ZIN(4) )
C
C*****COMPUTE PROJECTION DISTANCE, EQUATIONS ( 69 ) AND ( 71 ).
C
      D = NX * (AVX-XIN(1)) + NY * (AVY-YIN(1)) + NZ * (AVZ-ZIN(1))
      PD = ABS (D)
C
C*****EQUATIONS (73) AND (74).
C
      T = SQRT ( T1X * T1X + T1Y * T1Y + T1Z * T1Z )
      T1X = T1X / T
      T1Y = T1Y / T
      T1Z = T1Z / T
C
C*****EQUATION ( 75 ).
C
      T2X = NY * T1Z - NZ * T1Y
      T2Y = NZ * T1X - NX * T1Z
      T2Z = NX * T1Y - NY * T1X
C
C*****COMPUTE COORDINATES OF CORNER POINTS IN REFERENCE COORD. SYSTEM.
C*****EQUATION ( 72 )
C
C
C                          ---------- * ----------
C
      DO 1ØØØ J = 1, 4
      XP = XIN (J) + NX * D
      YP = YIN (J) + NY * D
      ZP = ZIN (J) + NZ * D
      D = -D
      XDIF = XP - AVX
      YDIF = YP - AVY
      ZDIF = ZP - AVZ
C
C*****TRANSFORM CORNER POINTS TO ELEMENT COORDINATE SYSTEM ( XI, ETA )
C*****WITH AVERAGE AS ORIGIN. EQUATION ( 8Ø ).
C
      XI(J) = T1X * XDIF + T1Y * YDIF + T1Z * ZDIF
 1ØØØ ETA(J)= T2X * XDIF + T2Y * YDIF + T2Z * ZDIF
C
C                          ---------- * ----------
C
C
C*****COMPUTE CENTROID, EQUATION ( 81 ).
C
      XIØ = Ø.3333333EØ * ( XI(4) * ( ETA(1) - ETA(2) ) + XI (2)
     1      * ( ETA(4) - ETA(1) ) ) / ( ETA(2) - ETA(4) )
      ETAØ=-Ø.3333333EØ * ETA(1)
C
C*****OBTAIN CORNER POINTS IN SYSTEM WITH CENTROID AS ORIGIN, EQ. ( 82
C
      DO 1Ø2Ø J = 1, 4
      XI (J) = XI (J) - XIØ
 1Ø2Ø ETA(J) = ETA(J) - ETAØ
C
C*****COMPUTATION AIDS.
C
      ETA2M1 = ETA (2) - ETA (1)
      ETA3M2 = ETA (3) - ETA (2)
      ETA4M3 = ETA (4) - ETA (3)
```

```
      ETA1M4 = ETA (1) - ETA (4)
      XI1M2  =  XI (1) -  XI (2)
      XI2M3  =  XI (2) -  XI (3)
      XI3M4  =  XI (3) -  XI (4)
      XI4M1  =  XI (4) -  XI (1)
      ETA2P4 = ETA (2) + ETA (4)
      XI3M1  =  XI (3) -  XI (1)
      XI4M2  =  XI (4) -  XI (2)
      ETA2M4 = ETA (2) - ETA (4)
      XI1234 =  XI (1) +  XI (2) + XI (3) + XI (4)
C
C*****TRANSFORM CENTROID TO REFERENCE COORDINATE SYSTEM, EQ. ( 83 ).
C
C
C                          ---------- * ----------
C
      XCENT = AVX + T1X * XIØ + T2X * ETAØ
      YCENT = AVY + T1Y * XIØ + T2Y * ETAØ
      ZCENT = AVZ + T1Z * XIØ + T2Z * ETAØ
C
C*****COMPUTE LARGER DIAGONAL VECTOR, EQUATION ( 84 ).
C
      TSQ = AMAX1 ( XI3M1 ** 2, XI4M2 ** 2 + ETA2M4 ** 2 )
      T = SQRT ( TSQ )
C
C*****COMPUTE AREA, EQUATION ( 85 ).
C
      AREA = Ø.5 * XI3M1 * ETA2M4
C
C*****COMPUTE SECOND MOMENTS IXX, IXY, IYY, EQS. ( 86 ) TO ( 88 ).
C
      IXX = 8.333333E-2 * XI3M1 * ( ETA (1) * XI4M2 * XI1234 + ETA2M4
     1    * ( XI (1) * ( XI (1) + XI (3) ) + XI (3) ** 2 ) + XI (2)
     2    * ETA (2) * ( XI1234 - XI (4) ) - XI (4) * ETA (4)
     3    * (XI1234 - XI (2) ) )
      IXY = 4.166667E-2 * XI3M1 * ( 2.Ø * XI (4) * ( ETA (1) ** 2 -
     1      ETA (4) ** 2 ) - 2.Ø * XI (2) * ( ETA (1) ** 2 - ETA (2)**2)
     2      + ( XI(1) + XI(3) ) * ETA2M4 * ( 2.Ø * ETA (1) + ETA2P4 ) )
      IYY = 8.333333E-2 * XI3M1 * ETA2M4 * ( (ETA (1) + ETA2P4) ** 2
     1    - ETA (1) * ETA2P4 - ETA (2) * ETA (4) )
C
C*****COMPUTE CONSTANTS FOR EQUATIONS ( 42 ) AND ( 43 ). ALSO EQ. ( 45 )
C
      D12SQ = XI1M2 ** 2 + ETA2M1  ** 2
      D12 = SQRT ( D12SQ )
      D23SQ = XI2M3 ** 2 + ETA3M2 ** 2
      D23 = SQRT ( D23SQ )
      D34SQ = XI3M4 ** 2 + ETA4M3 ** 2
      D34 = SQRT ( D34SQ )
      D41SQ = XI4M1 ** 2 + ETA1M4 ** 2
      D41 = SQRT ( D41SQ )
C
C                          ---------- * ----------
C
      XC ( KONTRL ) = XCENT
      YC ( KONTRL ) = YCENT
      ZC ( KONTRL ) = ZCENT
      UNITVX ( KONTRL ) = T1X
      UNITVY ( KONTRL ) = T1Y
      UNITVZ ( KONTRL ) = T1Z
      DO 15ØØ K = 1, 4
      XIN ( K ) = XIN ( K ) * FC
      YIN ( K ) = YIN ( K ) * FC
      ZIN ( K ) = ZIN ( K ) * FC
 15ØØ CONTINUE
C
C
C*****PRINT RESULTS -- SECTION 9.4 THE FIRST OUTPUT.
C
```

```
CS    NOTE THAT THE COMPRESSIBILITY CORRECTION IS BEING MADE HERE.
       XO      = XCENT * FC
       YO      = YCENT * FC /SUNBET
       ZO      = ZCENT * FC  /SUNBET
       TT      = T     * FC /SUNBET
       DD      = PD    * FC /SUNBET
       AR      = AREA * FC * FC /(SUNBET*SUNBET)
CT     IF ( NPRT .GE. 11 ) GO TO 175Ø
       IF(NPRT.GT.38) GO TO 175Ø
       NPRT = NPRT + 1
       IF ( I .EQ. 1 ) GO TO 176Ø
CT     WRITE ( 6, 51Ø ) I, XIN, NX, XO, DD, YIN, NY, YO, TT, ZIN,
CT    1                 NZ, ZO, AR
       WRITE ( 6, 5Ø2 ) I,XO,YO,ZO,NX,NY,NZ,DD,TT,AR
       GO TO 177Ø
 175Ø NPRT = Ø
       CALL HEADER
CT     WRITE ( 6, 53Ø )
       WRITE ( 6,  5ØØ )
CT76Ø WRITE ( 6, 52Ø ) N, I, XIN, NX, XO, DD, BTYPE, YIN, NY, YO,
CT    1                 TT, ZIN, NZ, ZO, AR
 176Ø WRITE(6,5Ø1) N,I,XO,YO,ZO,NX,NY,NZ,DD,TT,AR,BTYPE
 177Ø XN ( KONTRL ) = NX
       YN ( KONTRL ) = NY
       ZN ( KONTRL ) = NZ
       A  ( KONTRL ) = AREA
C
C**** WRITE 28 QUANTITIES ON TAPE 4 AS ONE LOGICAL RECORD.
C
 2ØØØ WRITE(4) XCENT, YCENT, ZCENT, T1X, T1Y, T1Z, T2X, T2Y, T2Z, NX,
      1          NY, NZ, XI(1), ETA(1), XI(2), ETA(2), XI(3), ETA(3),
      2          XI(4), ETA(4), TSQ, AREA, IXX, IXY, IYY, D12, D23,
      3          D34, D41

       ICOUNT = ICOUNT + MMIN
       RETURN
       END
```

```
      SUBROUTINE BSETUP
C
C     THIS SUBROUTINE SETS UP ALL THE NECESSARY PARAMETERS AND
C     REQUIREMENTS BEFORE CALLING THE BOUNDARY LAYER PROGRAM. ALSO
C     COMPUTES THE NEW SIGMA'S, THEN LOOPS BACK TO SUBROUTINE COMFLO
C     TO COMPUTE THE FINAL SOLUTIONS.
C          NOTE HERE THAT THE NECESSARY CHANGES FOR THE INTRODUCTION
C           OF VISCOUS EFFECTS ON THE NON-LIFTING BODY ALSO IS
C             INCLUDED. ALSO, SOME INPUT MODIFICATIONS ARE MADE.
      DIMENSION BLIND(2)
      DIMENSION           NTYPE  (  5Ø  ),     NLINE  (  5Ø  )
      DIMENSION           TITLE  (  18  ),     NSORCE (  1Ø  ) ,NSORNL(3Ø)
      DIMENSION           NWAKE  (  1Ø  ),     NSTRIP (  1Ø  )
      DIMENSION           IGN (  5Ø,1Ø  ),     IG1 (  5Ø,1Ø  )
      DIMENSION           KOUNT  (  5Ø  ),     NLT    (  5ØØ )
      DIMENSION           UNITVX ( 11ØØ ),     UNITVY ( 11ØØ )
      DIMENSION           UNITVZ ( 11ØØ ),     VELX   ( 11ØØ )

      DIMENSION           VELY   ( 11ØØ ),     VELZ   ( 11ØØ )
      DIMENSION           XCENT  ( 11ØØ ),     YCENT  ( 11ØØ )
      DIMENSION           ZCENT  ( 11ØØ ),     XC     (  1ØØ )
      DIMENSION           YC     (  1ØØ ),     ZC     (  1ØØ )
      DIMENSION           XB     (  1ØØ ),     YB     (  1ØØ )
      DIMENSION           ZB     (  1ØØ ),     V      (  1ØØ )
      DIMENSION           DEL    (  1ØØ ),     SM     (  1ØØ )
      DIMENSION           XINP   (  5ØØ ),     YINP   (  5ØØ )
      DIMENSION           ZINP   (  5ØØ ),     XLINP  (  5Ø  )
      DIMENSION           YLINP  (  5Ø  ),     ZLINP  (  5Ø  )
      DIMENSION           RHS    ( 11ØØ ),     DQDS   (  1ØØ )
      DIMENSION           VEL    ( 11ØØ ),     DQS    (  1ØØ )
      DIMENSION           VB     (  1ØØ ),     D      (  1ØØ )
      DIMENSION           T      (  1ØØ ),BT(1ØØ),    S      (  1ØØ )
      DIMENSION           DELS   (  1ØØ ),     IXFLAG (  1Ø  )
      DIMENSION           PCHORD (  51  ),     SK     (  51  )
      DIMENSION           VK     (  51  ),     XK     (  51  )
      DIMENSION           ZK     (  51  ),     DELK   (  51  )
      DIMENSION           DF     (  51  ),     TF     (  51  )
      DIMENSION           BETAK  (  51  )
C
      COMMON  / CONFLG / TITLE, CASE, LIFSEC, LIST, NOFF, IOUT, KONTRL,
     1                   KUTTA, MPR
      COMMON  / INFORM / NSORCE, NWAKE, NSTRIP, IG1, IGN
      COMMON  / CTABLE / XCENT, YCENT, ZCENT ,AMACH
      COMMON  / IMPORT / NON, NFLOW
      COMMON  / BDINFO / KOUNT, NLINE, NTYPE, NLT, ISECT
      COMMON  / VBLOCK / VELX, VELY, VELZ, VEL
      COMMON  / UVETOR / UNITVX, UNITVY, UNITVZ
      COMMON  / RSPARM / I2, SM, VB, DEL, DQS
      COMMON  / NLINP  / NL, XINP, YINP, ZINP
      COMMON  / LTINP  / LT, XLINP, YLINP, ZLINP
      COMMON/BL12/TI,RMI,UI,RI,PR,PRT,FK,RL,RMUI,RHOI,PSI,HE
     1           ,UE(1ØØ),RØ(1ØØ),TW(1ØØ),QW(1ØØ),RP(1ØØ),FW(1ØØ)
     2           ,BR(1ØØ),TE(1ØØ),RHOE(1ØØ),RMUE(1ØØ),GW(1ØØ),GPW(1ØØ)
     3           ,RF1(1ØØ),RF2(1ØØ),YS (1ØØ),IGX1(1ØØ),FPW(1ØØ),ROL(1ØØ
      COMMON  / EXTRA  / IXFLAG, SKIP, METHOD
      COMMON / APPA / NKK,NFF,MAT,IERR,KPRINT
      COMMON / NLFVIS / NLFSTR , NSORNL , NONLIF
C
      DATA                ZERO  / Ø.ØEØ /
      DATA BLIND/2HLO,2HUP/
      DATA                IONE / 1 /
      DATA           IRSIDE /  16  /
      DATA    PCHORD / Ø.ØØØØØØ, Ø.ØØ2273, Ø.ØØ4555, Ø.ØØ6867, Ø.ØØ9244,
     1       Ø.Ø11738, Ø.Ø14384, Ø.Ø17271, Ø.Ø2Ø464, Ø.Ø24Ø46, Ø.Ø28182,
     2       Ø.Ø32724, Ø.Ø38ØØ7, Ø.Ø44Ø48, Ø.Ø5Ø945, Ø.Ø58796, Ø.Ø67697,
     3       Ø.Ø77742, Ø.Ø89Ø22, Ø.1Ø1623, Ø.115622, Ø.131Ø92, Ø.148Ø98,
```

```
     4        0.166692, 0.186918, 0.208807, 0.232376, 0.257630, 0.284555,
     5        0.313123, 0.343287, 0.374981, 0.408119, 0.442592, 0.478271,
     6        0.515000, 0.552599, 0.590864, 0.629560, 0.668427, 0.707171,
     7        0.745470, 0.782969, 0.819281, 0.853981, 0.886611, 0.916674,
     8        0.943639, 0.966931, 0.985936, 1.000000  /
C
C
C
      SUNBET=SQRT(1.-AMACH*AMACH)
   10 FORMAT   ( 1H0, 35X, 'INTERMEDIATE PRINT IN SUBROUTINE SETUP'
     .'  (DIMENSIONS IN FEET)')
   20 FORMAT   ( 1H0, 41X, 'SECTION NO. = ', I2, 4X, 'TYPE = ', I1, 4X,
     1          'TOTAL STRIPS = ', I3 )
   30 FORMAT   ( 1H0,16X, '8L. STRIP', 3X, 'C. POINT', 9X, 'X', 15X,
     1          'Y', 15X, 'Z', 15X, 'V', 15X, 'S'                / 17X,
     2          9(1H-), 3X, 8(1H-), 5( 3X, 13(1H-) )    )
   32 FORMAT   ( 1H0, 8X, 'STRIP NO.', 3X, 'C. POINT', 9X, 'X', 15X,
     1          'Z', 15X, 'V', 15X, 'S', 14X, 'DEL', 13X, 'DQDS' / 9X,
     2          9(1H-), 3X, 8(1H-), 6( 3X, 13(1H-) )    )
   40 FORMAT   ( 1H , 19X, A5, 10X, I2, 6X, E13.6, 4( 3X, E13.6 ) )
   99 FORMAT   ( 1H , 16X, A5, 10X, I2, 6X, E13.6, 4( 3X, E13.6 ) )
   41 FORMAT   ( 1H , 11X, I2, 10X, I2, 3X,6F16.6)
   50 FORMAT   ( 1H0, 53X, 'THE NEW RIGHT - HAND SIDE' / )
   51 FORMAT   ( 1H0, 24X, 1HK, 10X, 6HPCHORD, 10X, 8HXK ( K ), 9X,
     1          8HZK ( K ), 9X, 8HSK ( K ), 9X, 8HVK ( K ) / 24X,
     2          3H---, 9X 6(1H-), 10X, 8(1H-), 3( 9X, 8(1H- ) ) / )
   52 FORMAT   ( 1H , 22X, I3, 5( 3X, F14.6 ) )
   53 FORMAT   ( 1H0, 37X, 1HI, 9X, 8HSM ( I ), 11X, 8HVB ( I ), 10X,
     1          10HDEL   ( I ) / 37X, 3H---, 3( 5X, 14(1H-) ) / )
   54 FORMAT   ( 1H0, 37X, 1HK, 9X, 8HSK ( K ), 11X, 8HVK ( K ), 10X,
     1          10HBETAK( K ) / 37X, 3H---, 3( 5X, 14(1H-) ) / )
   55 FORMAT   ( 1H , 36X, I3, 3( 5X, E14.6 ) )
   60 FORMAT   ( 1H , 7X, 8E15.6 )
   70 FORMAT   ( 1H0, 14X, 'STRIP NO. = ', I2, 3X, 'STAGNATION PANEL ='
     1          I4, 3X, 'X0 = ', E13.6, 3X, 'Y0 = ', E13.6, 3X,
     2          'Z0 = ', E13.6 )
   80 FORMAT   ( 1H0, 46X, 'NO SEPARATION CAN BE FOUND FOR THIS STRIP')
   92 FORMAT   ( 1H0, 46X, 'TOTAL POINT IN SETUP NOT EQUAL TO TOTAL',
     1                     'CONTROL POINT, PROGRAM ENDS FOR CORRECTION'
   94 FORMAT   ( 1H0, 47X, 'ELEMENTS INPUT TO B.L. PROGRAM = ', I5 /46X
     1          'ELEMENTS RETURN TO CALLING PROGRAM = ', I5 / 61X,
     1          'PROGRAM STOPS' / )
   95 FORMAT   ( 2F15.0, 3I5 )
   96 FORMAT   ( 1H0, 47X, 37HBOUNDARY LAYER PROGRAM OUTPUT FOLLOWS / )
   97 FORMAT   ( I5 )
C
C     FIRST, SET UP SOME COUNTERS
C               LS -- LIFTING SECTION COUNT.
C               NOS -- NON LIFTING SECTION COUNT.
C               KK -- ELEMENT COUNT
C               L1 -- INITIAL STRIP COUNT.
C               L2 -- FINAL   STRIP COUNT
C               KR -- ORDER FOR ARRANGING THE S"S, Q"S, DQ / DS"S.
C               KN -- OVERALL STRIP COUNT
C
      IERR   = 0
      IM     = 51
      KSTRIP = 0
      LS     = 0
      KK     = 0
      L1     = 1
      KTOTAL = 0
      KN     = 0
      NL     = 0
      READ  ( 5, 95 ) UI, RI, ISM, IBETA, NOCAL
C
C     UI --- REFERENCED VELOCITY  ( FT / SEC )
C     RI --- REYNOLD"S NUMBER PER FOOT.
C     ISM --  0, NO SMOOTHING.
```

```
C      IBETA -- Ø, NO SPECIAL BETA CALCULATION.
C      NOCAL -- NON-ZERO, NO DELTA STAR GENERATED FOR NON-LIFTING SEC.
C
C
C      TAKE ONE SECTION AT A TIME, PROCEED TO PREPARE FORMING OF A
C      BOUNDARY LAYER STRIP.
C
       DO 15ØØ IS = 1, ISECT
       JTYPE  = NTYPE ( IS )
       JSTRIP = NLINE ( IS )
       L2     = L1 + JSTRIP - 1
       IF ( JTYPE .EQ. 1 ) GO TO 5ØØ
C
C      JTYPE IS NON-LIFTING
C
       KSTART = Ø
       K1=L1
       LP=Ø
C       LS=LS+1
       NS=NLFSTR
C       IPOINT=NSORNL(NOS)
       REWIND 13
       DO 3ØØ LK = L1, L2
       NL     = NL + 2
       KN     = KN + 1
C       KSORCE = IABS ( NLT ( KN ) )
       IF(IS.EQ.1.OR.LK.NE.L1) KN=KN
       IF(IS.NE.1.AND.LK.EQ.L1) KN=KN+1
       LL=JSTRIP
       KSORCE = NSORNL(KN)
C      TYPE*,'KSORCE,JSUND',KSORCE,JSUND
       J1     = KK + 1
       J2     = KK + KSORCE
       IF(IS.LE.3) NOCAL=1
       IF(IS.GT.3) NOCAL=Ø
       IF ( NOCAL .GT. Ø ) GO TO 24Ø
       READ   ( 5, 97 ) NXT
       I2     = KSORCE + 1
       VB (1) = Ø.
       DELS(1)= Ø.
       SM  (1)= DELS (1)
C
C------ THE FOLLOWING MODIFICATIONS ARE DONE TO INCLUDE VISCOUS EFFECTS
C                ON THE NON-LIFTING BODY ALSO.
C
       DO 11111 KKK=1,KSORCE
        KK=KK+1
       B11=UNITVX(KK)
       B12=UNITVY(KK)
       B13=UNITVZ(KK)
       BVX=VELX(KK)
       BVY=VELY(KK)
       BVZ=VELZ(KK)
       V(KKK)=VEL(KK)
       XC(KKK)=XCENT(KK)
       YC(KKK)=YCENT(KK)
       ZC(KKK)=ZCENT(KK)
       BT(KKK)=B11*BVX+B12*BVY+BVZ*B13
11111    CONTINUE
       KMM1=KSORCE-1
       DO 11112 KKK=1,KMM1
       BTPROD=BT(KKK)*BT(KKK+1)
       IF(BTPROD.GE.ZERO) GO TO 11112
       KBSTRT=KKK
       IF(KBSTRT.LT.(KSORCE/2))LLLL=1
       IF(KBSTRT.EQ.(KSORCE/2))LLLL=Ø
       IF(LLLL.EQ.1)KBSTRT=KSORCE/2
       GO TO 11114
11112       CONTINUE
```

```
C--------------N O   S T A G N A T I O N   P O I N T   D E T E R M I N E D
      WRITE(6,8Ø)
      GO TO 4ØØØ
11114       CONTINUE
      BDENOM=BT(KBSTRT+1)-BT(KBSTRT)
      IF(LLLL.EQ.Ø) GO TO 12458
      XI=XC(KBSTRT)
      YI=YC(KBSTRT)
      ZI=ZC(KBSTRT)
      GO TO 13457
12458       XI= (BT(KBSTRT+1)*XC(KBSTRT)-BT(KBSTRT)*XC(KBSTRT+1))/BDENOM
      YI= (BT(KBSTRT+1)*YC(KBSTRT)-BT(KBSTRT)*YC(KBSTRT+1))/BDENOM
      ZI= (BT(KBSTRT+1)*ZC(KBSTRT)-BT(KBSTRT)*ZC(KBSTRT+1))/BDENOM
13457   CONTINUE
      DO 11115 IBBL=1,2
      CALL HEADER
      WRITE(6,1Ø)
      WRITE(6,2Ø) IS,JTYPE,JSTRIP
      YI=YI/SUNBET
      ZI=ZI/SUNBET
      WRITE(6,7Ø) LK,KBSTRT,XI,YI,ZI
      YI=YI*SUNBET
      ZI=ZI*SUNBET
      REWIND 13
      IBB=IBBL
      XB(1)=XI
      YB(1)=YI
      ZB(1)=ZI
      IF(IBB.EQ.2) GO TO 11116
      I2=KBSTRT+1
      TYPE*,'I2',I2
      KM=1
      DO 11117 K=2,I2
      KL=I2-KM
      XB(K)=XC(KL)
      YB(K)=YC(KL)
      ZB(K)=ZC(KL)
      VB(K)=V(KL)
      KM=KM+1
11117   CONTINUE
      GO TO 11118
11116       CONTINUE
      I2=KSORCE-KBSTRT+1
      TYPE*,'I2 FOR IBB=2',I2
      KM=KBSTRT
      DO 11119  K=2,I2
      KL=KM+1
      XB(K)=XC(KL)
      YB(K)=YC(KL)
      ZB(K)=ZC(KL)
      KM=KM+1
11119       CONTINUE
11118       DO 11109 KJ=2,I2
      DELX    = ( XB ( KJ ) - XB ( KJ-1 ) ) ** 2
      DELY    = ( YB ( KJ ) - YB ( KJ-1 ) ) ** 2
      DELZ    = ( ZB ( KJ ) - ZB ( KJ-1 ) ) ** 2
      DELS (KJ) = SQRT ( DELX + DELY + DELZ )
      SM    (KJ) = SM ( KJ-1 ) + DELS ( KJ )
11109   CONTINUE
C      XB (1) = Ø.5 * ( XINP ( NL ) + XINP ( NL-1 ) )
C      YB (1) = Ø.5 * ( YINP ( NL ) + YINP ( NL-1 ) )
C      ZB (1) = Ø.5 * ( ZINP ( NL ) + ZINP ( NL-1 ) )
C      WRITE ( 6, 1Ø )
C      WRITE ( 6, 2Ø ) IS, JTYPE, JSTRIP
C      XB(1)=XB(1)
C      YB(1)=YB(1)/SUNBET
C      ZB(1)=ZB(1)/SUNBET
C      WRITE ( 6, 7Ø ) LK, KSTART, XB (1), YB (1), ZB (1)
C      XB(1)=XB(1)
```

```
C       YB(1)=YB(1)*SUNBET
C       ZB(1)=ZB(1)*SUNBET
C       DO 1ØØ KJ = 2, I2
C       KK      = KK + I
C       XB (KJ)= XCENT ( KK )
C       YB (KJ)= YCENT ( KK )
C       ZB (KJ)= ZCENT ( KK )
C       VB (KJ)= VEL   ( KK )
C  1ØØ CONTINUE
      WRITE ( 6, 3Ø )
      DO 1Ø2 I = 1, I2
      XB(I)=XB(I)
      YB(I)=YB(I)/SUNBET
      ZB(I)=ZB(I)/SUNBET
      SM(I)=SM(I)
      WRITE ( 6, 4Ø ) BLIND(IBBL),I,XB (I),YB (I),ZB(I), VB (I), SM (I)
      XB(I)=XB(I)
      YB(I)=YB(I)*SUNBET
      ZB(I)=ZB(I)*SUNBET
      SM(I)=SM(I)
 1Ø2 CONTINUE
      IF ( ISM .GT. Ø ) GO TO 1Ø4
      IM      = I2
      DO 1Ø3 I = 1, IM
      SK (I) = SM (I)
      VK (I) = VB (I)
      XK (I) = XB (I)
      ZK (I) = ZB (I)
 1Ø3 CONTINUE
      GO TO 15Ø
 1Ø4 IF ( I2 .GT. 5Ø ) GO TO 14Ø
      DO 12Ø K = 1, IM
      SK (K) = SM (I2) * PCHORD (K)
 12Ø CONTINUE
C
      IR=I2-1
      CALL WAC ( I2, SM, XB, IM, SK, XK, DF, TF )
      CALL WAC ( I2, SM, ZB, IM, SK, ZK, DF, TF )
      CALL WAC ( I2, SM, VB, IM, SK, VK, DF, TF )
      IF(KPRINT.EQ.Ø) GO TO 15Ø
      WRITE ( 6, 51 )
      DO 13Ø K = 1, IM
      ZK(K)=ZK(K)/SUNBET
      WRITE ( 6, 52 ) K, PCHORD (K), XK (K), ZK (K), SK (K), VK (K)
       ZK(K)=ZK(K)*SUNBET
 13Ø CONTINUE
      GO TO 15Ø
 14Ø IM     = I2
 15Ø IF ( IBETA .EQ. Ø ) GO TO 18Ø
      BETAK (1) = 1.Ø
      IM1    = IM - 1
      IM2    = IM - 2
      DO 16Ø I = 2, IM1
      DI     = -(SK(I+1)-SK(I)) / ( (SK(I+1)-SK(I-1)) * (SK(I)-SK(I-1)))
      FI     =  (SK(I)-SK(I-1)) / ( (SK(I+1)-SK(I-1)) * (SK(I+1)-SK(I)))
      EI     = -( DI + FI )
      DVDS   = DI * VK (I-1) + EI * VK (I) + FI * VK (I+1)
      BETAK (I) = ( SK (I) / VK (I) ) * DVDS
 16Ø CONTINUE
      P1     = SK (IM) - SK (IM1)
      P2     = SK (IM) - SK (IM2)
      P3     = SK (IM1) - SK (IM2)
      DI     = P1 / ( P3 * P2 )
      FI     = P2 / ( P3 * P1 )
      EI     = ( 2.Ø*SK(IM) - SK(IM1) - SK(IM2) ) / ( P1 * P2 )
      DVDS   = DI * VK(IM2) - FI * VK(IM1) + EI * VK(IM)
      BETAK (IM) = ( SK (IM) / VK (IM) ) * DVDS
      WRITE ( 6, 54 )
      DO 165 I = 1, IM
```

```
      DO 21Ø I = 1, KSORCE
      ZC(I)=ZC(I)/SUNBET
      WRITE ( 6, 41 ) LK, I, XC (I), ZC (I), V (I), S (I),
     1                D (I), DQDS (I)
      ZC(I)=ZC(I)*SUNBET
  21Ø CONTINUE
      IF ( NOCAL .EQ. Ø ) GO TO 248
  24Ø DO 242 K = 1, KSORCE
      DQDS (K) = Ø.Ø
      D (K) = Ø.Ø
  242 CONTINUE
C        THIS PARTICULAR ASSIGNMENT OF KK=KK+KSORCE IS TO INCORPORATE
C          THE ELEMENT COUNT WHILE THE NON-LIFTING PART IS NOT TAKEN
C            FOR VISCOUS CALCULATIONS.THIS IS DISCARDED HERE.
C       KK = KK + KSORCE
  248 CONTINUE
C
C     PUT THE COMPUTED RHS TERMS IN THE ORDER OF THE CONTROL POINTS.
C
      KR      = Ø
      DO 25Ø J = J1, J2
      KR      = KR + 1
      RHS (J)= DQDS ( KR )
  25Ø CONTINUE
      KTOTAL = KTOTAL + KSORCE
C
C     INCREMENT TO THE NEXT STRIP
C
  3ØØ CONTINUE
      L1      = L2 + 1
      GO TO 15ØØ
C
C     THE FOLLOWING PART IS FOR THE LIFTING STRIPS.
C
5ØØ       IERR    = Ø
      IM      = 51
      KSTRIP = Ø
      KN      = Ø
      NL      = Ø
      K1      = L1
      LP      = Ø
      LS=Ø
      LS=LS+1
      NS      = NSTRIP ( LS )
      IEXTRA = IXFLAG ( LS )
      KWAKE  = NWAKE  ( LS )
      IPOINT = NSORCE ( LS )
      NP      = KWAKE + IPOINT
      IF ( IEXTRA .EQ. Ø ) GO TO 53Ø
      IF ( IEXTRA .EQ. 1  .OR.  IEXTRA .EQ. 3 ) GO TO 51Ø
      K2      = L2 + 2
      GO TO 55Ø
  51Ø K2      = L2 + 1
      GO TO 55Ø
  53Ø K2      = L2
  55Ø DO 13ØØ LK = L1, L2
      KN      = KN + 1
      LP      = LP + 1
      IPOINT = NSORCE ( LS )
      LIG1    = IG1 ( LP, LS )
      LIGN    = IGN ( LP, LS )
      NIGNOR = LIGN - LIG1
      IF ( LIGN .EQ. Ø .AND. LIG1 .EQ. Ø ) GO TO 6ØØ
      NIGNOR = NIGNOR + 1
  6ØØ KSORCE = IPOINT - NIGNOR
      J1      = KK + 1
      J2      = KK + KSORCE
      VB (1) = Ø.
      DELS(1)= Ø.
```

```
      SM  (1)= DELS (1)
      DO 650 K = 1, KSORCE
      KK     = KK + 1
      A11    = UNITVX ( KK )
      A12    = UNITVY ( KK )
      A13    = UNITVZ ( KK )
      VX     = VELX   ( KK )
      VY     = VELY   ( KK )
      VZ     = VELZ   ( KK )
      V  (K) = VEL    ( KK )
      XC (K) = XCENT  ( KK )
      YC (K) = YCENT  ( KK )
      ZC (K) = ZCENT  ( KK )
      T  (K) = A11 * VX + A12 * VY + A13 * VZ
  650 CONTINUE
      KM1    = KSORCE - 1
      DO 700 K = 1, KM1
      TPROD  = T ( K ) * T ( K+1 )
      IF ( TPROD .GE. ZERO ) GO TO 700
      KSTART = K
      GO TO 720
  700 CONTINUE
C
C     NO SEPARATION OF FLOWS, WRITE A MESSAGE AND QUIT.
C
      WRITE ( 6, 80 )
      GO TO 4000
C
C     FOLLOWING STATEMENTS ARE FOR GOOD RUNS.
C
  720 CONTINUE
      DENOM  = T ( KSTART+1 ) - T ( KSTART )
      XI     = ( T ( KSTART+1 ) * XC ( KSTART ) -
     1           T ( KSTART ) * XC ( KSTART+1 ) ) / DENOM
      YI     = ( T ( KSTART+1 ) * YC ( KSTART ) -
     1           T ( KSTART ) * YC ( KSTART+1 ) ) / DENOM
      ZI     = ( T ( KSTART+1 ) * ZC ( KSTART ) -
     1           T ( KSTART ) * ZC ( KSTART+1 ) ) / DENOM
C
C     XI,YI,ZI ARE COORDINATES OF THE INITIAL POINT ON BOTH LIFTING
C     BOUNDARY LAYER STRIPS.
C
      VB (1) = 0.
      DO 1200 IBL = 1, 2
      READ   ( 5, 97 ) NXT
      CALL HEADER
      WRITE ( 6, 10 )
      WRITE ( 6, 20 ) IS, JTYPE, JSTRIP
      XI=XI
      YI=YI/SUNBET
      ZI=ZI/SUNBET
      WRITE ( 6, 70 ) LK, KSTART, XI, YI, ZI
      YI=YI*SUNBET
      ZI=ZI*SUNBET
      REWIND 13
      IB     = IBL
      XB (1) = XI
      YB (1) = YI
      ZB (1) = ZI
      IF ( IB .EQ. 2 ) GO TO 800
      I2     = KSTART + 1
      KM     = 1
      DO 750 K = 2, I2
      KL     = I2 - KM
      XB (K) = XC ( KL )
      YB (K) = YC ( KL )
      ZB (K) = ZC ( KL )
      VB (K) = V  ( KL )
      KM     = KM + 1
```

```
  75Ø CONTINUE
       GO TO 91Ø
  8ØØ CONTINUE
       I2      = KSORCE - KSTART + 1
       KM      = KSTART
       DO 9ØØ K = 2, I2
       KL      = KM + 1
       XB (K) = XC ( KL )
       YB (K) = YC ( KL )
       ZB (K) = ZC ( KL )
       VB (K) = V  ( KL )
       KM      = KM + 1
  9ØØ CONTINUE
C
C      CALL CALCBL TO CALCULATE ARC LENGTHS AND CALL BL. PROGRAM.
C
  91Ø DO 915 K = 2, I2
       DELX    = ( XB ( K ) - XB ( K-1 ) ) ** 2
       DELY    = ( YB ( K ) - YB ( K-1 ) ) ** 2
       DELZ    = ( ZB ( K ) - ZB ( K-1 ) ) ** 2
       DELS(K)= SQRT ( DELX + DELY + DELZ )
       SM  (K)= SM ( K-1 ) + DELS ( K )
  915 CONTINUE
       WRITE ( 6, 3Ø )
       DO 92Ø I = 1, I2
       YB(I)=YB(I)/SUNBET
       ZB(I)=ZB(I)/SUNBET
       WRITE(6,99) BLIND(IBL), I, XB(I), YB(I), ZB(I), VB(I), SM(I)
       YB(I)=YB(I)*SUNBET
       ZB(I)=ZB(I)*SUNBET
  92Ø CONTINUE
       IF ( ISM .GT. Ø ) GO TO 928
       IM      = I2
       DO 924 I = 1, IM
       SK (I) = SM (I)
       VK (I) = VB (I)
       XK (I) = XB (I)
       ZK (I) = ZB (I)
  924 CONTINUE
       GO TO 95Ø
  928 IF ( I2 .GT. 5Ø ) GO TO 94Ø
       DO 93Ø K = 1, IM
       SK (K) = SM (I2) * PCHORD (K)
  93Ø CONTINUE
       IR      = I2 - 1
C
       CALL WAC ( I2, SM, XB, IM, SK, XK, DF, TF )
       CALL WAC ( I2, SM, ZB, IM, SK, ZK, DF, TF )
       CALL WAC ( I2, SM, VB, IM, SK, VK, DF, TF )
C
       IF(KPRINT.EQ.Ø)  GO TO 95Ø
       WRITE ( 6, 51 )
       DO 932 K = 1, IM
       ZK(K)=ZK(K)/SUNBET
       WRITE ( 6, 52 ) K, PCHORD (K), XK (K), ZK (K), SK (K), VK (K)
       ZK(K)=ZK(K)*SUNBET
  932 CONTINUE
       GO TO 95Ø
C
  94Ø IM      = I2
  95Ø IF ( IBETA .EQ. Ø ) GO TO 97Ø
       BETAK (1) = 1.Ø
       IM1     = IM - 1
       IM2     = IM - 2
       DO 955 I = 2, IM1
       DI      = -(SK(I+1)-SK(I)) / ((SK(I+1)-SK(I-1)) * (SK(I)-SK(I-1)))
       FI      =  (SK(I)-SK(I-1)) / ((SK(I+1)-SK(I-1)) * (SK(I+1)-SK(I)))
       EI      = -( DI + FI )
       DVDS    = DI * VK (I-1) + EI * VK (I) + FI * VK (I+1)
```

```
      WRITE ( 6, 55 ) I, SK (I), VK (I), BETAK (I)
  165 CONTINUE
  180 CONTINUE
C
      WRITE ( 13 ) IM, IBETA
      IF ( IBETA .GT. Ø ) WRITE ( 13 ) ( BETAK (K), K = 1, IM )
      WRITE ( 13 ) ( XK (K), K = 1, IM )
      WRITE ( 13 ) ( ZK (K), K = 1, IM )
      WRITE ( 13 ) ( VK (K), K = 1, IM )
      WRITE ( 13 ) ( SK (K), K = 1, IM )
      WRITE ( 13 ) NXT
      REWIND 13
C
      IF(KPRINT.GT.Ø)       WRITE ( 6, 96 )
      CALL BOUNDL
      REWIND 2
      READ ( 2 ) IK
      READ ( 2 ) ( DELK (K), K = 1, IK )
      REWIND 2
      IF ( IK .EQ. IM ) GO TO 182
      WRITE ( 6, 94 ) IM, IK
      STOP
  182 CONTINUE
      IF ( ISM .EQ. Ø ) GO TO 184
C
      CALL WAC ( IK, SK, DELK, I2, SM, DEL, DF, TF )
      GO TO 186
  184 DO 185 K = 1, IM
      DEL (K) = DELK (K)
  185 CONTINUE
      IF(KPRINT.EQ.Ø) GO TO 189
  186 WRITE ( 6, 53 )
      DO 188 K = 1, I2
      WRITE ( 6, 55 ) K, SM (K), VB (K), DEL (K)
  188 CONTINUE
  189      CONTINUE
C
      IR=KSORCE
      CALL CALCRS
C
C      DO 2ØØ K = 2, I2
C      S (K-1)= SM ( K )
C      D (K-1)= DEL ( K )
C      V (K-1)= VB( K )
C      DQDS (K-1) = DQS ( K )
C  2ØØ CONTINUE
      GO TO (11120,11130),IBB
11120      KM=KBSTRT
      DO 11121 K=2,I2
      S(KM)=SM(K)
      D(KM)=DEL(K)
      V(KM)=VB(K)
      DQDS(KM)=DQS(K)
      KM=KM-1
11121      CONTINUE
        GO TO 11115
11130      KM=KBSTRT+1
      DO 11122 K=2,I2
      S(KM)=SM(K)
      D(KM)=DEL(K)
      V(KM)=VB(K)
      DQDS(KM)=DQS(K)
      KM=KM+1
11122      CONTINUE
11115      CONTINUE
      CALL HEADER
      IF(KPRINT.GT.Ø)  WRITE ( 6, 5Ø )
        IF(KPRINT.GT.Ø)    WRITE ( 6, 6Ø ) ( DQDS (I), I = I, KSORCE )
      WRITE ( 6, 32 )
```

```
      BETAK (I) = ( SK (I) / VK (I) ) * DVDS
  955 CONTINUE
      P1      = SK (IM) - SK (IM1)
      P2      = SK (IM) - SK (IM2)
      P3      = SK (IM1) - SK (IM2)
      DI      = P1 / ( P3 * P2 )
      FI      = P2 / ( P3 * P1 )
      EI      = ( 2.Ø*SK(IM) - SK(IM1) - SK(IM2) ) / ( P1*P2 )
      DVDS    = DI * VK(IM2) - FI * VK(IM1) + EI * VK(IM)
      BETAK (IM) = ( SK (IM) / VK (IM) ) * DVDS
      WRITE ( 6, 54 )
      DO 965 I = 1, IM
      WRITE ( 6, 55 ) I, SK (I), VK (I), BETAK (I)
  965 CONTINUE
  97Ø CONTINUE
C
      WRITE ( 13 ) IM, IBETA
      IF ( IBETA .GT. Ø ) WRITE ( 13 ) ( BETAK (K), K = 1, IM )
      WRITE ( 13 ) ( XK (K), K = 1, IM )
      WRITE ( 13 ) ( ZK (K), K = 1, IM )
      WRITE ( 13 ) ( VK (K), K = 1, IM )
      WRITE ( 13 ) ( SK (K), K = 1, IM )
      WRITE ( 13 ) NXT
      REWIND 13
C
      IF(KPRINT.GT.Ø) WRITE ( 6, 96 )
      CALL BOUNDL
      REWIND 2
      READ ( 2 ) IK
      READ ( 2 ) ( DELK (K), K = 1, IK )
      REWIND 2
      IF ( IK .EQ. IM ) GO TO 1ØØØ
      WRITE ( 6, 94 ) IM, IK
      STOP
 1ØØØ CONTINUE
      IF ( ISM .EQ. Ø ) GO TO 1Ø1Ø
C
      CALL WAC ( IK, SK, DELK, I2, SM, DEL, DF, TF )
      GO TO 1Ø2Ø
C
 1Ø1Ø DO 1Ø15 K = 1, IM
      DEL (K) = DELK (K)
 1Ø15 CONTINUE
 1Ø2Ø CONTINUE
      IF(KPRINT.EQ.Ø) GO TO 1Ø26
      WRITE ( 6, 53 )
      DO 1Ø25 K = 1, I2
      WRITE ( 6, 55 ) K, SM (K), VB (K), DEL (K)
 1Ø25 CONTINUE
 1Ø26 CONTINUE
C
      CALL CALCRS
C
      IR      = KSORCE
C
C     PUT THE S"S, DEL"S BACK TO ORDER.
C
      GO TO ( 1Ø5Ø, 11ØØ ), IB
 1Ø5Ø KM      = KSTART
      DO 1Ø6Ø K = 2, I2
      S (KM) = SM ( K )
```

```
      D (KM) = DEL ( K )
      V (KM) = VB ( K )
      DQDS (KM) = DQS ( K )
      KM     = KM - 1
 1060 CONTINUE
      GO TO 1200
 1100 KM     = KSTART + 1
      DO 1150 K = 2, I2
      S (KM) = SM ( K )
      D (KM) = DEL ( K )
      V (KM) = VB ( K )
      DQDS (KM) = DQS ( K )
      KM     = KM + 1
 1150 CONTINUE
 1200 CONTINUE
C
C
C
      CALL HEADER
      IF(KPRINT.GT.0) WRITE ( 6, 50 )
      IF(KPRINT.GT.0) WRITE ( 6, 60 ) ( DQDS (I), I = 1, KSORCE )
      WRITE ( 6, 32 )
      DO 1220 I = 1, KSORCE
      ZC(I)=ZC(I)/SUNBET
      WRITE ( 6, 41 ) LK, I, XC (I), ZC (I), V (I), S (I),
     1                D (I), DQDS (I)
      ZC(I)=ZC(I)*SUNBET
 1220 CONTINUE
C
C     PLACE THE COMPUTED RHS'S IN THE ORDER OF THE CONTROL POINTS.
C
      KR     = 0
      DO 1250 J = J1, J2
      KR     = KR + 1
C
      RHS (J)= DQDS ( KR )
C
 1250 CONTINUE
      KTOTAL = KTOTAL + KSORCE
 1300 CONTINUE
      L1     = L2 + 1
 1500 CONTINUE
      IF ( KTOTAL .EQ. KONTRL ) GO TO 2000
      WRITE ( 6, 92 )
      GO TO 4000
C
C     PUT THE NEW RHS COLUMN ON UNIT IRSIDE.
C
 2000 WRITE ( IRSIDE ) IONE
      WRITE ( IRSIDE ) ( RHS ( I ), I = 1, KONTRL )
      WRITE ( 6, 50 )
      WRITE ( 6, 60 ) ( RHS ( I ), I = 1, KONTRL )
      REWIND IRSIDE
```

```
      RETURN
4000  IERR   = 1
      RETURN
      END
```

```
CSC*DECK PRINT
C       OVERLAY(LINK,6,2)
C      LIST,NONE
C       PROGRAM PRINT
        SUBROUTINE PRINT
C
C*****THIS SUBROUTINE PRINTS OUT THE FINAL OUTPUT WHICH HAS THE MOST
C*****IMPORTANT DATA.
C
      LOGICAL        INTIAL,   LAST ,    SKIP
      REAL           MOMENX,   MOMENY,   MOMENZ
      REAL           MXI     ( 500  ),    MYJ    ( 500  )
      REAL           MZK     ( 500  )
      DIMENSION      COMSIG  ( 1100 ),    AREA   ( 1100 )
      DIMENSION      VX      ( 1100 ),    VY     ( 1100 )
      DIMENSION      VZ      ( 1100 ),    VSUM   ( 1100 )
      DIMENSION      VEL     ( 1100 ),    DCX    ( 1100 )
      DIMENSION      DCY     ( 1100 ),    DCZ    ( 1100 )
      DIMENSION      VN      ( 1100 ),    PCOEF  ( 1100 )
      DIMENSION      NSORCE  (   10 ),    NWAKE  (   10 )
      DIMENSION      IG1 ( 50, 10 ),      IGN ( 50, 10 )
      DIMENSION      NLINE   (   50 ),    NSTRIP (   10 )
      DIMENSION      NLT     (  500 ),    NTYPE  (   50 )
CT    DIMENSION      NLINE1  (   10 ),    NLINEN (   10 )
      DIMENSION      KSPLIT  (   10 ),    KOUNT  (   50 )
      DIMENSION      FSTRPX  (  500 ),    FSTRPY (  500 )
      DIMENSION      FSTRPZ  (  500 ),    FSECX  (   50 )
      DIMENSION      FSECY   (   50 ),    FSECZ  (   50 )
      DIMENSION      SECMOX  (   50 ),    SECMOY (   50 )
      DIMENSION      XC      ( 1100 )    ,     YC      ( 1100 )
      DIMENSION      ZC      ( 1100 )    ,     XN      ( 1100 )
      DIMENSION      YN      ( 1100 )    ,     ZN      ( 1100 )
      DIMENSION      SECMOZ  (   50 )
      DIMENSION      ALPHAX  ( 10 ) ,     ALPHAY ( 10 )
      DIMENSION      ALPHAZ  (   10 ),    TITLE  (   18  )
C
C                          ---------- * ----------
C
      COMMON  / CONFLG / TITLE, CASE, LIFSEC, LIST, NOFF, IOUT, KONTRL,
     1                   KUTTA, MPR
      COMMON  / INFORM / NSORCE, NWAKE, NSTRIP, IG1, IGN
      COMMON  / BDINFO / KOUNT, NLINE, NTYPE, NLT, ISECT
      COMMON  / IMPORT / NON, NFLOW
      COMMON  / VBLOCK / VX, VY, VZ, VEL
      COMMON  / FINAL  / COMSIG, VSUM, DCX, DCY, DCZ, VN, PCOEF, B(50)
      COMMON  / EXTRA  / IXFLAG ( 10 ), SKIP
      COMMON  / ORIGIN / ORIGNX, ORIGNY, ORIGNZ
      COMMON  / ANGLE  / MANGLE, ALPHAX, ALPHAY, ALPHAZ  ,SUNBET
      COMMON  / ATTACK / ANGLEX, ANGLEY, ANGLEZ
      COMMON  / OUTPUT / FPRINT, IANGLE
      COMMON  / CTABLE / XC, YC, ZC ,AMACH
      COMMON  / NORMAL / XN, YN, ZN
      COMMON  / ATABLE / AREA
      COMMON  / SCALE  / FC, BL
C
C                          ---------- * ----------
C
```

```
 100  FORMAT(1H0,50X,32HON - BODY  POINTS  FINAL  OUTPUT  /
     1        1H ,50X,32(1H-))
 110  FORMAT(1H ,11X,4HN   M,5X,2HX0,8X,2HY0,8X,2HZ0,8X,2HVX,
     1       8X,2HVY,8X,2HVZ,8X,2HCP,8X,2HVN,5X,5HSIGMA)
 120  FORMAT(1H ,9X,2I3,9F10.6)
 130  FORMAT(1H ,12X,I3,9F10.6)
 140  FORMAT(1H ,8X,3(20X,12H************))
 150  FORMAT(1H ,29X,20HSTRIP    (  FORCE/Q ),3(2X,E13.6))
 155  FORMAT(1H ,29X,20HSECTION  (  FORCE/Q ),3(2X,E13.6))
 158  FORMAT(1H ,29X,20HBODY     (  FORCE/Q ),3(2X,E13.6))
 160  FORMAT(1H ,29X,20HSTRIP    ( MOMENT/Q ),3(2X,E13.6))
 162  FORMAT(1H ,29X,20HSECTION  ( MOMENT/Q ),3(2X,E13.6))
 164  FORMAT(1H ,29X,20HBODY     ( MOMENT/Q ),3(2X,E13.6))
C
 200 FORMAT ( 1H0, 52X, 28HKUTTA  POINTS  FINAL  OUTPUT /
     1          1H , 52X, 28 ( 1H- ) )
 210 FORMAT ( 1H0, 32X, 5HKUTTA, 7X, 2HX0, 11X, 2HVX, 11X, 2HVT, 11X,
     1          3HDCX, 11X, 2HNX / 1H , 32X, 6HPOINTS, 6X, 2HY0, 11X,
     2          2HVY, 11X, 2HVN, 11X, 3HDCY, 11X, 2HNY / 1H , 44X, 2HZ0,
     3          11X, 2HVZ, 11X, 2HCP, 11X, 3HDCZ, 11X, 2HNZ / 1H , 32X,
     4          6(1H-), 3X, 9(1H-), 4( 4X, 9(1H-) ) )
 220 FORMAT ( 1H0, 33X, I4, 5F13.6 / ( 38X, 5F13.6 ) )
 230 FORMAT ( 1H0, 51X, 9HSTRIP NO., 10X, 11HB ( STRIP ) /
     1          1H , 51X, 9(1H-),  9X, 13(1H-) / )
 240 FORMAT ( 1H , 53X, I4, 12X, E13.6 )
 300 FORMAT ( 1H0, 49X, 33HOFF - BODY  POINTS  FINAL  OUTPUT  /
     1          1H , 50X, 33( 1H- ) )
 310 FORMAT ( 1H0, 38X, 6HPOINTS, 7X, 2HX0, 11X, 2HVX, 12X, 2HVT,
     1 10X, 3HDCX,9X,'UOFOUT' / 1H , 51X, 2HY0, 11X, 2HVY, 11X, 4HVTSQ,
     2  9X, 3HDCY,9X,'VOFOUT' / 1H , 51X, 2HZ0, 11X, 2HVZ, 12X, 2HCP,
     3          10X, 3HDCZ,9X,'WOFOUT' / 1H , 38X, 6(1H-), 5( 4X, 9(1H-)
     4          ))
 320 FORMAT ( 1H0, 39X, I3, 2X, 5F13.6 / ( 45X, 5F13.6 ) )
 330 FORMAT ( 1H0, 40X, I4,28H ON - BODY  POINTS  MISSED. ,
     1          22HEXECUTION  TERMINATED. )
 340 FORMAT ( 1H0, 41X, I4, 24H KUTTA  POINTS  MISSED. ,
     1          22HEXECUTION  TERMINATED. )
 350 FORMAT ( 1H0, 40X, I4, 27H OFF-BODY  POINTS  MISSED. ,
     1          22HEXECUTION  TERMINATED. )
C
C                        ---------- * ----------
C
C*****OUTPUT INCLUDES 3 PARTS--
C     (1) THE ON-BODY POINTS FINAL OUTPUT,
C     (2) THE KUTTA POINTS FINAL OUTPUT, AND
C     (3) THE OFF-BODY POINTS FINAL OUTPUT
C
C
C      KT25W - PRESSURE DATA OUTPUT FLAG (0 TO SUPPRESS)
C
      KT25W=1
      SKIP   = .FALSE.
C
      CALL HEADER
      WRITE(6,100)
      WRITE(6,110)
      NPRT = 3
C
      L1 = 1
      LS = 0
      KK = 0
      KM1 = NFLOW - MANGLE
      KN = 0
      FORCEX = 0.
      FORCEY = 0.
      FORCEZ = 0.
      MOMENX = 0.
      MOMENY = 0.
      MOMENZ = 0.
```

```
      FC2    = FC * FC
      DO 36Ø NY = 1, 5ØØ
      MXI ( NY ) = Ø.
      MYJ ( NY ) = Ø.
      MZK ( NY ) = Ø.
      FSTRPX ( NY ) = Ø.
      FSTRPY ( NY ) = Ø.
  36Ø FSTRPZ ( NY ) = Ø.
      DO 362 NR = 1, 5Ø
      SECMOX ( NR ) = Ø.
      SECMOY ( NR ) = Ø.
      SECMOZ ( NR ) = Ø.
      FSECX ( NR ) = Ø.
      FSECY  ( NR ) = Ø.
  362 FSECZ  ( NR ) = Ø.
C
C*****NPRT -- PRINT-LINE CONTROL COUNT
C*****L1 ---- INITIAL STRIP COUNT
C*****LS ---- LIFTING SECTION INCREMENT FLAG
C*****KK ---- CONTROL ELEMENT INCREMENT COUNT
C
      PPPP=Ø.
      DO 1ØØØ IS = 1, ISECT
      J1SUND=NTYPE(IS)
      J2SUND=NLINE(IS)
      J3SUND=KOUNT(IS)
      WRITE(6,9923)
 9923      FORMAT(2X,'THE CONFIGURATION IS AFDL.24,MACH=Ø.925')
CS      PUNCH 9923
      WRITE(7,9923)
      WRITE(6,5ØØØ) IS,J1SUND,J3SUND,J2SUND
C      PUNCH 5ØØ2,IS,J1SUND,J3SUND,J2SUND
      WRITE(7,5ØØ2) IS,J1SUND,J3SUND,J2SUND
 5ØØØ      FORMAT(1HØ,11X,'BODY SECTION NO=', I2,1ØX,'TYPE = 'I1,
     1      1ØX,'TOTAL NO OF POINTS=', I4,1ØX, 'NO. OF STRIPS =', I3 )
 5ØØ2      FORMAT(2X,'BODY SECTION NO=',I3,3X,'TYPE=',I3,3X,'TOTAL NUMBER O
     1      F  POINTS=',I5,3X,'NUMBER OF STRIPS=',I3)
      JTYPE = NTYPE ( IS )
      IF ( JTYPE .EQ. Ø ) GO TO 38Ø
      LP = Ø
      LS = LS + 1
      IPOINT = NSORCE ( LS )
      NXTRA = IXFLAG ( LS )
      IF ( NXTRA .EQ. Ø ) GO TO 369
      IF ( NXTRA - 2 ) 364, 366, 368
  364 INTIAL = .TRUE.
      LAST  = .FALSE.
      GO TO 37Ø
  366 INTIAL = .TRUE.
      LAST  = .TRUE.
      GO TO 37Ø
  368 INTIAL = .FALSE.
      LAST  = .TRUE.
      GO TO 37Ø
  369 INTIAL = .FALSE.
      LAST  = .FALSE.
  37Ø JSTRIP = NSTRIP ( LS )
      GO TO 385
  38Ø JSTRIP = NLINE ( IS )
  385 L2 = L1 + JSTRIP - 1
      DO 9ØØ LK = L1, L2
      KN = KN + 1
      IF ( JTYPE .EQ. Ø ) GO TO 42Ø
      IF ( ( LK .EQ. L1 ) .AND. INTIAL ) GO TO 9ØØ
      IF ( ( LK .EQ. L2 ) .AND. LAST   ) GO TO 9ØØ
      LP = LP + 1
      LIG1 = IG1 ( LP, LS )
      LIGN = IGN ( LP, LS )
      NIGNOR = LIGN - LIG1
```

```
      IF ( LIG1 .EQ. Ø .AND. LIGN .EQ. Ø ) GO TO 4ØØ
      NIGNOR = NIGNOR + 1
  4ØØ KSORCE = IPOINT - NIGNOR
      GO TO 5ØØ
  42Ø KSORCE = IABS ( NLT ( KN ) )
  5ØØ DO 8ØØ K = 1, KSORCE
      KK = KK + 1
      XØ = XC ( KK )
      YØ = YC ( KK )
      ZØ = ZC ( KK )
      XNORM = XN ( KK )
      YNORM = YN ( KK )
      ZNORM = ZN ( KK )
C
C     CALCULATE THE FORCE COMPONENTS.
C
      CPAREA = -PCOEF (KK) * AREA (KK)
      FX     = CPAREA * XNORM * FC2
      FY     = CPAREA * YNORM * FC2
      FZ     = CPAREA * ZNORM * FC2
      FSTRPX ( LK ) = FSTRPX ( LK ) + FX
      FSTRPY ( LK ) = FSTRPY ( LK ) + FY
      FSTRPZ ( LK ) = FSTRPZ ( LK ) + FZ
C
C     CALCULATE THE MOMENT COMPONENTS.
C
      RX     = XØ * FC - ORIGNX
      RY     = YØ * FC - ORIGNY
      RZ     = ZØ * FC - ORIGNZ
      XMO = RY * FZ - RZ * FY
      YMO = RZ * FX - RX * FZ
      ZMO = RX * FY - RY * FX
      MXI ( LK ) = MXI ( LK ) + XMO
      MYJ ( LK ) = MYJ ( LK ) + YMO
      MZK ( LK ) = MZK ( LK ) + ZMO
      XO     = XØ * FC
      YO     = YØ * FC /SUNBET
      ZO     = ZØ * FC /SUNBET
      AR     = AREA ( KK ) * FC * FC
6ØØ1      FORMAT(7F1Ø.5,I2)
      IF ( NPRT .GE. 39 ) GO TO 6ØØ
      NPRT = NPRT + 1
      IF ( K .EQ. 1 ) GO TO 7ØØ
      SUND1=VX(KK)
      SUND2=VY(KK)
      SUND3=VZ(KK)
      CALL CPSUND(XØ,YØ,ZØ,SUND1,SUND2,SUND3,SUNX,SUNY,SUNZ)
      WRITE ( 6, 13Ø ) K, XO,YO,ZO,SUNX,SUNY,SUNZ,PCOEF(KK),
     I         VN(KK),COMSIG (KK)
C
      IF(KT25W.EQ.Ø) GO TO 8ØØ
      KKW=Ø
C
        GO TO 8ØØ
  6ØØ   NPRT = Ø
CS      CALL HEADER
CS      WRITE ( 6, 11Ø )
 7ØØ    CONTINUE
      WRITE ( 6, 12Ø ) LK, K, XO,YO,ZO,SUNX,SUNY,SUNZ,PCOEF(KK),
     1         VN(KK),COMSIG (KK)

C
      IF(KT25W.EQ.Ø) GO TO 8ØØ
      KKW=Ø
      IF(K.EQ.1) KKW=1
      IF(K.EQ.1 .AND. LK.EQ.1) KKW=4
C
  8ØØ CONTINUE
      IF ( NPRT .LT. 36 ) GO TO 82Ø
```

```
      NPRT = 0
      CALL HEADER
      WRITE ( 6, 110 )
  820 NPRT = NPRT + 4
      WRITE ( 6, 140 )
      WRITE ( 6, 150 ) FSTRPX (LK), FSTRPY (LK), FSTRPZ (LK)
      WRITE ( 6, 160 ) MXI ( LK), MYJ ( LK ), MZK ( LK )
      WRITE ( 6, 140 )
      FSECX (IS) = FSECX (IS) + FSTRPX (LK)
      FSECY (IS) = FSECY (IS) + FSTRPY (LK)
      FSECZ (IS) = FSECZ (IS) + FSTRPZ (LK)
      SECMOX (IS) = SECMOX (IS) + MXI (LK)
      SECMOY (IS) = SECMOY (IS) + MYJ (LK)
      SECMOZ (IS) = SECMOZ (IS) + MZK (LK)
  900 CONTINUE
      L1 = L2 + 1
      IF ( NPRT .LT. 37 ) GO TO 920
      NPRT = 0
      CALL HEADER
      WRITE ( 6, 110 )
  920 NPRT = NPRT + 4
      WRITE ( 6, 140 )
      WRITE ( 6, 155 ) FSECX (IS), FSECY (IS), FSECZ (IS)
      WRITE ( 6, 162 ) SECMOX (IS), SECMOY (IS), SECMOZ (IS)
      WRITE ( 6, 140 )
      FORCEX = FORCEX + FSECX (IS)
      FORCEY = FORCEY + FSECY (IS)
      FORCEZ = FORCEZ + FSECZ (IS)
      MOMENX = MOMENX + SECMOX (IS)
      MOMENY = MOMENY + SECMOY (IS)
      MOMENZ = MOMENZ + SECMOZ (IS)
 1000 CONTINUE
      IF ( NPRT .LT. 37 ) GO TO 1010
      NPRT = 0
      CALL HEADER
 1010 NPRT = NPRT + 4
      WRITE(6,140)
      WRITE ( 6, 158 ) FORCEX, FORCEY, FORCEZ
      WRITE ( 6, 164 ) MOMENX, MOMENY, MOMENZ
      WRITE ( 6, 140 )
      IF ( KK .NE. KONTRL ) GO TO 3200
      IF ( KUTTA .EQ. 0 ) GO TO 2100
C
C*****NOW THE KUTTA POINTS OUTPUT
C
      NPRT = 45
      DO 2000 K= 1, KUTTA
      KK = KK + 1
      X0 = XC ( KK )
      Y0 = YC ( KK )
      Z0 = ZC ( KK )
      XNORM = XN ( KK )
      YNORM = YN ( KK )
      ZNORM = ZN ( KK )
      IF ( NPRT .GE. 10 ) GO TO 1500
      NPRT = NPRT + 1
      GO TO 1800
 1500 NPRT = 0
      CALL HEADER
      WRITE ( 6, 200 )
      WRITE ( 6, 210 )
      X0=X0*FC
      Y0=Y0*FC/SUNBET
      Z0=Z0*FC/SUNBET
 1800 WRITE ( 6, 220 ) K, X0, VX( KK ), VEL( KK ), DCX( KK ), XNORM,
     1                  Y0, VY( KK ), VN( KK ), DCY( KK ), YNORM, Z0,
     2                  VZ( KK ), PCOEF( KK ), DCZ( KK ), ZNORM
      X0=X0/FC
      Y0=Y0/FC*SUNBET
```

```
      Z0=Z0/FC*SUNBET
 2000 CONTINUE
      WRITE ( 6, 140 )
C
C*****NEXT, THE OFF-BODY POINTS OUTPUT
C
 2100 IF ( NOFF .EQ. 0 ) GO TO 3100
CS       PUNCH 5003
      WRITE(7,5003)
 5003       FORMAT(5X,'THE FOLLOWING PUNCHED CARDS ARE FOR THE OFF BODY POIN
     1 TS')
      NPRT = 45
      KCOUNT = KK - KONTRL
      IF ( KCOUNT .NE. KUTTA ) GO TO 3400
      DO 3000 K = 1, NOFF
      KK = KK + 1
      X0 = XC ( KK )
      Y0 = YC ( KK )
      Z0 = ZC ( KK )
      IF ( NPRT .GE. 10 ) GO TO 2500
      NPRT = NPRT + 1
      GO TO 2800
 2500 NPRT = 0
      CALL HEADER
      WRITE ( 6, 300 )
      WRITE ( 6, 310 )
 2800       SUND4=VX(KK)
      SUND5=VY(KK)
      SUND6=VZ(KK)
      CALL CPOFF(X0,Y0,Z0,SUND4,SUND5,SUND6,UOFOUT,VOFOUT,WOFOUT)
      X0=X0*FC
      Y0=Y0*FC/SUNBET
      Z0=Z0*FC/SUNBET
       WRITE ( 6, 320 ) K, X0, VX( KK ), VEL( KK ), DCX( KK ),UOFOUT,Y0,
     1               VY( KK ), VSUM( KK ), DCY( KK ),VOFOUT, Z0, VZ( KK ),
     2                    PCOEF( KK ), DCZ( KK ),WOFOUT
      X0=X0/FC
      Y0=Y0/FC*SUNBET
      Z0=Z0/FC*SUNBET
 3000 CONTINUE
      WRITE ( 6, 140 )
      KCHECK = KK - KONTRL - KUTTA
      IF ( KCHECK .NE. NOFF ) GO TO 3600
 3100 CALL HEADER
      WRITE ( 6, 230 )
      DO 3110 K = 1, KM1
      WRITE ( 6, 240 ) K, B (K)
 3110 CONTINUE
      WRITE ( 6, 140 )
      GO TO 4000
 3200 MISS = IABS ( KK - KONTRL )
      WRITE ( 6, 330 ) MISS
      GO TO 4000
 3400 MISS = IABS ( KCOUNT - KUTTA )
      WRITE ( 6, 340 ) MISS
      GO TO 4000
 3600 MISS = IABS ( KCHECK - NOFF )
      WRITE ( 6, 350 ) MISS
 4000 RETURN
      END
```

INITIAL DISTRIBUTION

DTIC-DDAC	2
AUL/LSE	1
ASD/ENSZ	1
AFATL/DLODL	2
AFATL/CC	1
HQ USAF/SAMI	1
OO-ALC/MHWRB	1
AFIS/INT	1
HQ TAC/DRA	1
HQ USAFE/DOQ	1
HQ PACAF/DOQQ	2
HQ TAC/INAT	1
ASD/XRX	1
USA TRADOC SYS ANAL ACTY	1
COMIPAC (PT-2)	1
HQ PACAF/OA	1
USA BALLISTIC RESCH LAB	1
AFATL/CCN	1
AFATL/DLODA	1
ASD/ENESS	1
HQ TAC/XPS (STINFO)	1
6510 TEST GP/ENML	1
USMC/REDSTONE SCI INFO CNTR	2
NSC (TECH LIBRARY)	1
SANDIA NATIONAL LABS	1
NAVAL WPNS CNTR (CODE 3243)	1
AFATL/DLCA	4
DEPT OF AEROSPACE ENG INDIAN INST OF SCIENCE	2
UNIV OF TENNESSEE SPACE INSTITUTE/DR SUNDARAM	3
UNIV OF TENNESSEE SPACE INSTITUTE/TECH LIB	2

(The reverse of this page is blank)